CARL VON OSSIETZKY universität OLDENBURG

School of Computing Science, Business Administration, Economics, and Law
Department of Computing Science

Dynamic Coalition Formation in Electricity Markets

by

Sebastian Beer

submitted to the Department of Computing Science
in partial fulfillment of the requirements for the degree of
Doctor of Engineering
at the Carl von Ossietzky University of Oldenburg

Referees:

Prof. Dr. Michael Sonnenschein
Apl. Prof. Dr.-Ing. Jürgen Sauer
Prof. Dr. Sascha Ossowski

Date of disputation: December 16, 2016

To Maria

Abstract

The structure of the liberalized electricity industry can be distinguished into two complementary levels. On an economic level, different players trade power products at electricity markets in order to match their individual supply and demand. On a physical level, the stipulated electricity amounts are then produced by generation units and delivered to the consumers through a dedicated supply network. In this regard, throughout the last years several driving forces induced a restructuring of the formerly centralized supply system towards a decentralized provision. While on generation side conventional large-scale power stations are successively replaced by distributed energy resources like solar power or wind energy plants, on consumption side small-sized loads are integrated as active parts into the supply process based on innovative demand side management techniques.

However, the ongoing change led to a disintegration of the economic and physical level. Minimum volume thresholds of today's wholesale markets hinder small-scale energy units from a direct participation. Moreover, during operation many of the newly installed units are exposed to stochastic effects which cause deviations from contractually specified electricity amounts. Finally, the integration of distributed energy resources is subsidized in many countries. If these guaranteed payments run out in the future, the plants have to compete with large-scale units in the market and thus face an increased pressure to cover their generation costs.

Against this background, the thesis at hand proposes DYCE, a new agent-based approach for dynamic coalition formation in electricity markets. Coalitions allow small-scale production or consumption units to cumulate their potentials and fulfill power products which exceed their individual capabilities. In contrast to existing aggregation concepts like Virtual Power Plants, the proposed approach is fully distributed and takes both the preferences of the individual units and those of the global supply system into account. Moreover, by integrating topological information of the grid, it allows for the trade of localized power products which can be utilized by system operators for network management. Because the coalitions form for the fulfillment of a single product only, they are able to respond to changing conditions in a temporally flexible fashion. If a product was successfully traded, the gained surplus is divided among the coalition members in a fair way based on a game-theoretical distribution model.

Zusammenfassung

Die Struktur der liberalisierten Elektrizitätswirtschaft lässt sich grundsätzlich in zwei komplementäre Ebenen unterscheiden. So handeln auf einer ökonomischen Ebene unterschiedliche Teilnehmer elektrische Energie auf dedizierten Märkten, um Angebot und Nachfrage in Einklang zu bringen. Auf einer physikalischen Ebene werden die vertraglich vereinbarten Mengen dann von Generatoren erzeugt und über ein ausreichend dimensioniertes Übertragungs- und Verteilnetz an die Verbraucher geliefert. In diesem Zusammenhang haben in den letzten Jahren verschiedene Antriebsfaktoren zu einer anhaltenden Dezentralisierung des ursprünglich zentral strukturierten Versorgungssystems geführt. Diese äußert sich zum einen darin, dass auf Erzeugerseite konventionelle Großkraftwerke sukzessive durch Einheiten geringerer Leistung wie Solar- oder Windkraftanlagen ersetzt werden, die primär in das Nieder- und Mittelspannungsnetz einspeisen. Zum anderen erfolgt auf Verbraucherseite eine vermehrte Integration von Lasten mittels Techniken des Demand Side Managements, welche eine aktive Teilnahme am Versorgungsprozess ermöglichen.

Vor dem Hintergrund dieses Wandels werden die beschriebenen Strukturebenen allerdings nicht mehr den sich neu ergebenen Anforderungen gerecht. Einerseits stellen die an heutigen Elektrizitätsmärkten handelbaren Mindestmengen Eintrittsbarrieren für viele der neu integrierten Einheiten dar, welche eine direkte Teilnahme am Marktgeschehen verhindern. Weiterhin sind insbesondere nicht-steuerbare Erzeuger und Verbraucher wie Windkraftanlagen oder Waschmaschinen stochastischen Einflüssen ausgesetzt, die bei einer Produkterfüllung zu Abweichungen von vertraglich festgesetzten Energiemengen führen können. Schließlich wird in vielen Ländern die Integration von auf regenerativen Energiequellen beruhenden Anlagen staatlich subventioniert, so dass diese bei Auslaufen der Förderungen einem erhöhten Wettbewerbsdruck ausgesetzt sind.

Als Beitrag zur Lösung dieser Problemstellung wird in der vorliegenden Arbeit die agentenbasierte Methode DYCE vorgestellt, die es computergesteuerten Energieeinheiten ermöglicht, ihre technischen Potenziale zu aggregieren und Koalitionen für den Handel von Elektrizitätsprodukten zu bilden. Im Gegensatz zu bestehenden Ansätzen wie virtuellen Kraftwerken erfolgt der Formationsprozess dabei vollständig dezentral und in selbstorganisierter Art und Weise, wobei durch einen kombinierten Optimierungsansatz sowohl die Präferenzen der individuellen Einheiten als auch diejenigen des globalen Versorgungssystems berücksichtigt werden. Die Integration netztopologischer Informationen erlaubt darüber hinaus eine Erfüllung von lokalen Produkten, welche sich auf definierte Abschnitte im Netz beziehen und von Netzbetreibern zur Systemoptimierung eingesetzt werden können. Da die Formation einer Koalition grundsätzlich produktbezogen erfolgt, kann von den Einheiten zeitlich flexibel auf kurzfristige Ereignisse reagiert werden. Wurde der Handel eines Produkts erfolgreich abgeschlossen, wird abschließend der erwirtschaftete Mehrwert auf Basis eines spieltheoretischen Modells in fairer Art und Weise unter den Koalitionsmitgliedern verteilt.

Contents

Figures

Tables

Abbreviations

ACL	Agent Communication Language
AI	Artificial Intelligence
BPMN	Business Process Model and Notation
CCD	Central Composite Design
CCOR	Combined Coalition Regrouping
CFP	Call for Proposal
CHP	Combined Heat and Power
COPE	Combinatorial Optimization for Portfolio Enhancement
CSG	Coalition Structure Generation
CVPP	Commercial Virtual Power Plant
DAI	Distributed Artificial Intelligence
DER	Distributed Energy Resource
DOE	Design of Experiments
DSM	Demand Side Management
DSRG	Design Science Research Guideline
DSO	Distribution System Operator
DYCE	Dynamic Coalition Formation in Electricity Markets
DYCE-FM	DYCE Formal Model
EU	European Union
FIPA	Foundation for Intelligent Physical Agents
HC	Hard Constraint
ICT	Information and Communication Technology
MAS	Multiagent System
MEU	Maximum Expected Utility
OFAT	One-Factor-at-a-Time
KQML	Knowledge Query and Manipulation Language
RSM	Response Surface Modeling
SC	Soft Constraint
SGAE	Smart Grid Algorithm Engineering
SPADA	Set Partitioning Algorithm for Distributed Agents
TSO	Transmission System Operator
TVPP	Technical Virtual Power Plant
VPP	Virtual Power Plant
WEP	Wind Energy Plant

1 Introduction

With regard to its function to cover the electrical demand of customers, the structure of the liberalized electricity industry can be distinguished into two complementary levels. On an economic level, different players trade electrical energy at dedicated markets according to their individual needs. On a physical level, the traded amounts are then produced by generators and delivered to the consumers via a sufficiently dimensioned supply network. Because to date electricity is still not storable in considerable amounts, in the course of provision supply and demand have to be matched in each point in time. This fact puts specific organizational and technical requirements on both the economic and physical level which have to be fulfilled in order to guarantee a stable supply.
Until the 1990s, the electricity industry was characterized by a centralized supply paradigm where a few vertically integrated utilities governed all steps of the supply chain – from generation to transmission and distribution through to retail – and thus both of the described structural levels. As a result, electricity demand was mainly covered by large-scale power plants such as nuclear or coal-fired plants and delivered to consumers via a hierarchically organized supply network. Because these oligopolistic market structures were lacking competition and thus incentives for cost reductions, supply companies were typically strictly regulated by federal institutions in order to prevent the execution of market power.
However, in recent years several driving forces gave impetus to a restructuring of the conventional supply infrastructure. First, on the economic level, energy regulations were enacted in many countries in order to induce a liberalization of the electricity markets and introduce competition among the market players. These measures gave rise to the creation of bilateral and multilateral markets for the trade of electricity-related products according to defined market rules. In particular, derivatives and spot markets were established in order to provide participants an appropriate instrument to conclude long- and short-term contracts for the supply of electrical energy. Moreover, on the physical level, depleting fossil sources and a growing environmental awareness induced an increased integration of small-scale units into the power grid. Distributed energy resources (DERs) like solar power, combined heat and power (CHP) or wind energy plants (WEPs) were more and more incorporated into the supply infrastructure successively supplementing and substituting conventional fossil-fueled plants. In many countries, this trend is supported by political measures in order to meet long-term goals with respect to greenhouse gas reductions and pave the way towards a sustainable electricity provision. In particular, in 2009 the European Union (EU) enacted Directive 2009/28/EC [Dir09] prescribing binding target values for the member states with regard to shares of energy which have to be extracted from renewable sources, with an EU-wide overall goal of a 20 % share by 2020. In order to achieve this goal, in 2011 the German government adopted a national renewable energy action plan [NRE10] outlining measures and instruments to promote the use of renewable sources. These include the Renewable Energy Sources Act [Act00] which was already passed in 2000 and later adopted by many countries worldwide. In addition to the integration of DERs, in recent years a wide range of demand side man-

agement (DSM) techniques were conceived in order to utilize temporal flexibilities of consumption units and shift load into economically advantageous time intervals for a more efficient power provision (see e.g. [SLBT15]).
However, as both restructuring processes were originally induced by different motivations, the economic and physical level are not perfectly integrated any more which avoids a full exploitation of economic and ecological potentials.

1.1 Motivation and Requirements

Substantially shaping the electricity industry in the course of the last years, the restructuring processes outlined in the previous section generally allow for a decentralized electricity supply under competitive market conditions. However, in order to completely exploit associated potentials, the economic and physical level have still to be integrated with regard to the following aspects:

Barriers to entry Being designed with the conventional supply paradigm in mind, today's electricity markets are primarily aligned to large-scale power plants. I.e., the trading rules typically prescribe minimum volume thresholds which have to be met by actors in order to be allowed to participate in the market. Small-scale generation and consumption units are thus excluded from a direct participation because of their insufficient capabilities. Particularly for DERs which are typically characterized by small generation capacities, these thresholds constitute economic barriers to entry. Even in bilateral markets in which actors trade products individually according to their specific needs, similar problems arise when participants face offers exceeding their individual generation or consumption potentials.

Stochastic effects In contrast to the conventional, flexibly controllable power plants, many of the newly integrated production and consumption units (like wind energy plants or dishwashers) are exposed to stochastic effects which cause deviations from their forecasted operation schedules. In order to conform to the specified amounts of concluded contracts, these errors have to be compensated either by the unit owners before physical fulfillment through appropriate trades at the spot market or – in the worst case – by the transmission system operators during physical fulfillment through retrieval of expensive control reserve.

Topology-aware fulfillment The integration of distributed energy units into the power grid holds new potentials with regard to grid-related system services which allow to avoid the need for expensive grid expansions. For instance, localized power products can be used in order to prevent network congestions if appropriate economic incentives are provided to the market participants located in the corresponding grid section. Moreover, a decentralized electricity supply allows for a more efficient provision through a feed-in close to demand which reduces losses resulting from transmission or transformation. However, in order to exploit these kinds of potentials, grid-related information has to be incorporated into the design of an electricity market. In this

regard, current research investigates new types of markets which take related aspects into account [Wis16, BBRA10]. In contrast to conventional wholesale markets, these allow for a trade of topology-aware system services and drive the trend towards a more efficient provision.

Subsidized feed-in As already mentioned at the beginning of this chapter, in many countries the integration of DERs is subsidized through regulations. However, the guaranteed payments often run out at some point in the future (see for instance [Act00]). Then, today's subsidized plants will face an increased pressure to cover their generation costs as they have to compete with large-scale units in the market.

With regard to these issues, various concepts have been conceived and practically examined in the course of the last years. Common idea is the aggregation of small-scale energy units into a single pool, where a specific control mechanism is used in order to organize and coordinate the members according to a joint goal. In order to allow for an automated operation, the units are connected by means of information and communication technology (ICT). Such pools are for instance referred to as Virtual Power Plants (VPPs) [PRS07], cooperatives [CRK+11], or virtual devices [SLBT15].
With regard to electricity markets in general and the above mentioned problems in particular, an aggregation of energy units provides several advantages. First, by cumulating their generation or consumption potentials, it enables small-scale actors to meet minimum volume thresholds of wholesale markets and thus to trade products which exceed their individual capabilities. Second, by aggregating units with a different reliability or complementing capabilities (like producers and storage), it allows for a reduction or compensation of stochastic effects and thus for a more reliable product fulfillment saving members from high extra costs. Finally, through an aggregation and goal-directed coordination of its members, a pool allows for a more flexible response to changing market conditions as well as an increased market power enabling members to compete more successfully in the market.
Although basically having the same goals, current aggregation concepts often vary with regard to the internal organization and coordination of the integrated units. However, as common characteristics, most approaches published to date organize the members in a hierarchical relationship and coordinate them by means of a central control logic according to economic aspects, where the pool is typically assumed to be a fixed set which does not significantly change over time. However, when it comes to electricity trading in general and short-term-oriented spot markets in particular, these concepts have a range of drawbacks with regard to the following aspects:

Robustness The application of a central control logic for coordinating the units of a pool constitutes a single point of failure which may result in a malfunction or – in the worst case – a total breakdown of the system in case of errors. Even if robustness is introduced through redundancy, the cost for keeping mirrored systems in sync is typically very high.

Scalability If a set of units is to be divided into different pools according to a given optimization goal, centralized approaches typically fall short of scaling well with an increasing number of actors in terms of performance and efficiency. Consequently, the maximum manageable number of units is restricted by the technical capabilities of the central computing system.

Individual preferences In a competitive market, a central control of units which are contractually bound to a pool operator contradicts the intention of each market player to maximize its individual profit. I.e., the units are not able to flexibly apply their own trading strategies based on local optimization goals.

Temporal flexibility A temporally fixed composition of a pool impedes a flexible response to changing market conditions like price fluctuations. As a consequence, economic profits are lost which otherwise could be gained through the aggregation of a more beneficial set of units.

Trusted Cooperations If energy units belong to different owners, an aggregation requires some degree of trust among the members of a pool. Integrating related criteria into the formation process promotes the truthful behavior of the participants because harmful activities have negative consequences on the willingness of the others to cooperate in the future. However, to date most pooling concepts lack in concepts which permit trusted cooperations between different actors.

Payoff Distribution Because of the economic advantages described above, an aggregation of energy units allows for monetary benefits which can be distributed among the members of a pool. However, current concepts mainly consider the organization and operation of a pool but do not address the problem of dividing a resulting surplus according to specific criteria.

Topology awareness Finally, most pooling concepts proposed to date do not take topological aspects of the grid into account. In recent times, first approaches have been published which take spatial data of the units into account in order to allow for a provision of system services depending on this kind of information [PRS07, BSD+09]. However, there is yet no approach integrating topological information of all producers and consumers in the grid in order to enable a trade of localized power products and exploit the efficiency potentials as described above.

The above shortcomings motivate the development of a new aggregation method which is aligned to the current trends in the energy domain. In order to solve the discussed problems, we expect the approach to fulfill the following requirements:

- R_{DC} *(distributed control)*, i.e. to form pools in a completely decentralized fashion,
- R_S *(scalability)*, i.e. to scale well with an increasing number of units,
- R_{CO} *(combined optimization)*, i.e. to consider both local and global preferences,
- R_{TF} *(temporal flexibility)*, i.e. to allow for a dynamic reorganization over time,
- R_{TC} *(trusted cooperations)*, i.e. to take the trustworthiness of actors into account,
- R_{PD} *(payoff distribution)*, i.e. to allow for a fair division of a gained surplus, and
- R_{TA} *(topology awareness)*, i.e. to integrate grid-related information of the units.

The above requirements are based on the fundamental assumption that through a distributed design we obtain a robust and scalable aggregation method which enables participants to optimize their local profits while still preserving the possibility to form pools according to a global optimization goal. This assumption is supported by the successful application of decentralized systems in other domains which are characterized by inherent distributed processes or infrastructures, like manufacturing, transportation, or electronic commerce [JSW98]. By means of a combined optimization approach, we intend to take both the preferences of the individual participants and those of the global supply system into account.
In order to further emphasize the need for a new aggregation method, we next provide a more thorough overview of the state of the art before specifying the objectives of this thesis in detail.

1.2 Related Work

In what follows, we discuss research which is related to the context of this thesis. Throughout the descriptions, we refer to the previously stated requirements in order to point out the advantages and drawbacks of the considered approaches with regard to our addressed problem of forming pools of energy units in the context of electricity markets. As methods for the aggregation of different actors are also examined in the fields of e-commerce and distributed artificial intelligence (DAI), we do not restrict our discussion to the domain of electric power provision but include concepts from these areas as well. More precisely, research in the field of e-commerce and electronic markets considers the problem of how to form coalitions of customers in order to increase the profits of buyers and sellers through the trade of wholesale lots. Contrary, in the domain of distributed artificial intelligence related questions are typically considered as the problem of coalition structure generation (CSG) [RMWJ15] where a set of agents $A = \{a_1, \ldots, a_n\}$ is to be partitioned into a coalition structure $CS = \{C_1, \ldots, C_m\}$ – i.e. a set of exhaustive and disjunct coalitions $C_i \subseteq A$ – such that global value is maximized. More precisely, given two functions $v(C) \in \mathbb{R}$ and $V(CS) \in \mathbb{R}$ measuring the value of a coalition and a coali-

tion structure, respectively, goal of the optimization problem is to identify an optimal coalition structure

$$CS^* = \arg\max_{CS \in \widetilde{CS}} V(CS), \quad V(CS) = \sum_{C \in CS} v(C), \tag{1.1}$$

where $\widetilde{CS}$ denotes the set of all coalition structures, $\bigcup_{C \in CS} C = A$, and $\forall C, C' \in CS$ with $C \neq C' : C \cap C' = \emptyset$. As the CSG problem is NP-complete [SLA+99], related research particularly examines concepts which allow for an efficient approximation of an optimal solution.

1.2.1 Electric Power Provision

In [PRS07], Pudjianto, Ramsay, and Strbac discuss two related approaches for the aggregation of distributed energy resources and controllable loads referred to as commercial virtual power plant (CVPP) and technical virtual power plant (TVPP). In conjunction, both concepts are intended to provide a functionality similar to that of a conventional large-scale power plant connected to the transmission grid. A virtual power plant is here generally understood as a pool of units which is characterized by a single operating profile, where the latter is obtained through an aggregation of all unit-related parameter values which determine the final overall behavior (like outputs, generation limits, or ramp rates). As the participating units are potentially distributed over the whole grid, the capabilities of a VPP are furthermore affected by network-related properties such as impedances or losses. Based on this general notion, the authors propose the above mentioned specialized types CVPP and TVPP which serve the purposes of market participation and system management, respectively. More precisely, a commercial virtual power plant provides its members the advantage of a direct access to wholesale markets as well as the possibility to optimize their trading behavior. Moreover, it allows for a subgrouping of the participants by region if the market imposes specific constraints with regard to the units' locations. In contrast, a technical virtual power plant provides services for the management of the distribution and transmission grid. As these might be restricted to specific regions, a TVPP always aggregates units from the same local network. Because on level of the distribution grid a provision of system services requires individual information about the units (like the set operation schedules), these data are transferred by the CVPPs which include the concerned units as members. Contrary, for services on level of the transmission grid, all necessary information is aggregated into a single profile in order to describe the overall behavior of the pool and make it comparable to that of other large-scale power plants.

Although extending the typical functionality of a virtual power plant, CVPPs and TVPPs still do not fulfill most of our previously specified requirements. As both types are intended to be operated by single actors, they can be interpreted as centralized aggregation concepts which optimize the pool according to a single objective function and provide only a limited degree of scalability and organizational flexibility. Moreover, they do neither allow for trusted cooperations among participants nor for a distribution of a gained

surplus. However, as first approaches in the context of VPPs, both types integrate grid-related information which enables a spatial aggregation of units.
As another approach, in [KSK10] Kok, Scheepers, and Kamphuis describe PowerMatcher, an agent-based control concept for distributed energy units. Although originally developed for the purpose of supply and demand matching in electricity networks [KWK+05], the approach is also suitable for operating a cluster of units as a virtual power plant with respect to a desired goal. Its control functionality is realized through a multiagent system (MAS) in which software agents take on different roles according to their intended purpose. The organizational structure of the system is given by a tree which root represents a unique auctioneer agent being responsible for the price formation within a cluster. The auctioneer collects bids from local device agents which are either directly or indirectly connected to the root. As representative of an energy unit, each device agent strives for an economically optimal control of the very same taking into account all given constraints. The operational flexibilities which are exploitable for this task are generally determined by the type of the unit which can be one of a set of predefined categories (like stochastic operation devices, shiftable operation devices, or electricity storage devices). According to their control strategy, the device agents create bids in the form of demand functions which specify the power amounts they are willing to buy or sell at specific prices in a future time interval. These are submitted to the auctioneer agent which iteratively determines an equilibrium price based on the received information. After its assessment, the price is communicated back to the device agents which enables them to infer the amounts of power being allocated to their units. In order to achieve scalability, the approach allows to aggregate subsets of device agents by means of concentrator agents which form interfaces between the auctioneer and the subgroups. From the viewpoint of an auctioneer, a concentrator agent behaves like a single device agent by aggregating the submitted demand functions of the clustered agents into a single bid, while from the viewpoint of a subgroup it acts like an auctioneer by forwarding the assessed equilibrium prices. If a cluster is not intended to balance itself, an objective agent can be attached to the auctioneer which implements an alternative functionality. This way, the included units can for instance be operated as a virtual power plant which strives for the provision of a predefined operation schedule.
With regard to our requirements, PowerMatcher is designed as a distributed system, where scalability is achieved through an internal subgrouping of energy units. However, the hierarchical organization of a cluster is still prone to a malfunction or breakdown of single agents and does not dynamically change over time. Moreover, while trust-related issues are briefly discussed in [KWK+05], specific concepts are not explicitly integrated into the control process. Similarly, the provision of system services for network management is described as a potential functionality of a cluster in [KSK10], but no detailed information is given about a corresponding implementation of the objective agent or the grid-related requirements which have to be fulfilled by the cluster. Finally, while the approach is suitable for the operation of a VPP, the publications do not detail related aspects like specific objective functions or the distribution of a gained payoff.

As another agent-based approach, in [Lün12] Lünsdorf describes a self-organization[1] method for demand side management in mid and low voltage grids (a summary of the approach can also be found in [SLBT15]). Basically, the distributed heuristic aggregates controllable small-scale units like dishwashers or washing machines into virtual devices in order to alter the global load of a grid in a favourable way (for instance by reducing economically expensive load peaks). To this end, each substation is equipped with a software agent which controls the consumption units of the subordinate low voltage grid and coordinates its actions with other agents within a defined neighborhood. By including only those substations within a maximum range, the latter allows to restrict the activities of an agent to a limited section of the grid. In the course of an iterative optimization process, the agents then cluster their units into virtual devices with the goal to maximize global value. In each optimization cycle, an agent checks for each of its local units if it can be successfully allocated to a virtual device within a defined search space. The latter includes a subset of its own and neighboring virtual devices and is dynamically expanded in the course of the optimization process according to the expected probability to successfully assign a unit to an included device. This probability is generally assumed to be the higher, the higher the centrality of an agent controlling a device, i.e. the higher the total sum of units being controlled by the agent and its neighbors. With a rising number of iterations, the search space thus includes more and more devices of agents with a lower centrality. If in the course of an optimization cycle a unit can be successfully allocated to a new virtual device, the controlling agent updates its own data model and sends a notification to all affected neighbors in order to ensure a persistent system state. Contrary, if no allocation is possible, a new virtual device is created for the unit and assigned to the neighbor with the highest centrality in order to raise the probability that it soon becomes part of a bigger device. The optimization process finally ends when all participants have explored their whole neighborhood. However, if the load prognosis of a unit changes, the controlling agent reacts to the new situation and resumes its optimization activities.
With regard to our considered requirements, Lünsdorf's self-organization method allows for a distributed and temporally flexible creation of virtual devices for the purpose of a DSM measure which may be restricted to a specific region in the grid. As the heuristic is based on local knowledge and agents only consider an allocation of their own units in the course of the optimization process, it can also be expected to scale well with the overall number of units in the grid. However, as newly created devices are always assigned to agents with a high centrality, there is generally a bias with regard to the importance of different actors which decreases the robustness of the system. Moreover, the approach only considers a maximization of global value and does not address any aspects related to trustworthy cooperations or the distribution of a gained surplus.

[1] Self-organization refers to the capability of a system to change its organization without external control [DGK05].

1.2.2 Electronic Commerce

In [TSCY01], Tsvetovat et al. consider the problem of forming customer coalitions in the context of electronic markets in order to allow for volume discount prices resulting from a cumulated demand. After discussing the related economic incentives for both buyers and sellers, the authors present two protocols for coalition formation (referred to as post-negotiation and pre-negotiation protocol) which specify the basic interactions between all involved actors, i.e. a coalition leader, a number of potential coalition members, a set of suppliers, and a central coalition server. The proposed protocols generally differ in the order in which the actual formation process and the negotiations with the suppliers are conducted. In the context of the post-negotiation protocol, the coalition leader first initiates the formation of a coalition for the purchase of a specific item by using the coalition server which transfers a corresponding notification to the potential members. If these decide to join, they reply by sending a message to the initiator which starts a negotiation with the suppliers if either a maximum coalition size or a specified deadline has been met. After a deal has been made, the coalition leader finally collects the payments from the members and distributes the purchased items. Contrary, in the context of the pre-negotiation protocol, the contractual conditions are first negotiated with the suppliers which are then used as basis for the formation of a coalition.
With regard to our considered requirements, the presented protocols only allow for a centralized formation of coalitions by means of a mediating server, where these are built dynamically for the purchase of a single product only. Despite their centralized design, both approaches can be expected to scale well with the number of customers as no complex optimization problem has to be solved and the formation processes terminate if a specific coalition size or a deadline has been met. However, for the same reason the protocols can not be applied to more complex scenarios. Moreover, they integrate neither trust-related aspects nor a method for the distribution of a gained surplus.
A more sophisticated formation scheme termed GroupBuyAuction is presented by Yamamoto and Sycara in [YS01]. In contrast to the previously discussed protocols, the proposed approach allows customers to express their preferences more flexibly and distribute the final profit of a coalition in a stable way. I.e., before a formation process starts, buyers are initially able to specify their interest in multiple items of a considered product category (like cameras or notebooks) along with corresponding reservation prices. In the course of an auction, sellers then provide bids for the different products in the form of price schedules which reflect their volume discounts for different amounts of sold items. Based on the made bids, a group leader then partitions the buyers into subgroups by first identifying the most valuable coalition for any of the products and then proceeding recursively for the remaining buyers and items. The value of a coalition is generally defined by the gained profit, i.e. the difference between the summed reservation prices specified by its members and the total cost of the purchased products. Thus, the applied algorithm does not optimize the value of the whole buyer group but the one of the iteratively determined coalitions. After the formation process has ended, the group leader is also responsible for distributing the surplus of each coalition among

the members, where the resulting divisions are stable in terms of the game-theoretical concept of the core [SLB09]. The latter guarantees that no subset of members has an incentive to leave a coalition because of a higher profit gained when working on its own. With regard to our specified requirements, the proposed formation scheme represents a centralized but dynamic approach which restricts the organizational binding of coalition members to the trade of a single product only. As the running time of the included algorithm is $O(n \cdot \log n)$, with n being the number of buyers, the approach is applicable to high quantities of customers and further allows for a division of the resulting surplus. However, no trust-related aspects are integrated into the formation process. Moreover, the presented algorithm only improves the value of the coalitions and not that of the whole buyer group.

1.2.3 Distributed Artificial Intelligence

One of the first application-independent algorithms for agent-based coalition formation was proposed by Ketchpel in [Ket95]. The approach represents an iterative optimization process in which agents form coalitions through bilateral negotiations with the goal to maximize their individual payoff. Each negotiation round consists of four different steps referred to as Communication Phase, Calculation Phase, Offers Phase, and Unification Phase. As first step, an agent initially exchanges information with all other agents in the system which is required for assessing the benefit of a potential cooperation. The data are used in the Calculation Phase in order to determine the individual payoff which is associated with each bilateral collaboration, where the calculations are based on the game-theoretical concept of the Shapley value [SLB09]. The latter allows for a fair division of a jointly achieved benefit according to the contributions of the involved actors[2]. Based on the results, an agent next creates a preference list comprising its favoured cooperation partners in descending order. In the course of the third phase, it then negotiates with other agents by running through the list and making proposals for a collaboration while accepting or declining those received from others, where each agent is generally constrained to maintain a single pending offer at a time. If the interactions lead to an agreement, the involved agents finally form a coalition with a designated representative and act in the next round as a single entity. This way, coalitions iteratively grow until no more cooperation is possible and the algorithm eventually terminates.
With regard to our requirements, Ketchpel's heuristic allows for a decentralized and temporally flexible formation of coalitions according to a given goal. However, despite its distributed nature, it can only be expected to provide restricted scalability as in each negotiation round information is exchanged between all agents in the system. Moreover, because the latter are only allowed to maintain a single pending offer at a time and messages do not expire, the algorithm's running time may become unacceptably high if proposals accumulate over time. With regard the distribution of a coalition's value, the author notes that a division is implicity calculated in the course of the negotiations.

[2] If only two actors are involved, the individual shares are also referred to as bilateral Shapley values [CKY98].

However, from the descriptions it is not clear how a share should be further distributed if the negotiating party constitutes a coalition and not a single agent. Finally, the approach only considers a maximization of individual value and does not integrate aspects of trust. Alternatively, in [SK98] Shehory and Kraus address the problem of task allocation through decentralized coalition formation. More precisely, the authors consider multiagent systems in which each agent has a number of specific capabilities and a set of tasks has to be fulfilled. In order to solve a task cooperatively, agents are able to join forces and form coalitions. The related optimization problem is then analogous to that of coalition structure generation in the sense that the given tasks are to be allocated to coalitions in such a way that their summed value (resulting from task execution) is maximized (cf. Equation 1.1). Furthermore, the authors consider two special cases in which coalitions are allowed to overlap with regard to their members and the given tasks are associated with a defined precedence order. However, as these are not relevant for the context of this thesis, in the following we only discuss the algorithm proposed for solving the first problem. Basically, the presented heuristic consists of two iterative steps in which the agents first distributively calculate the values of all possible coalitions and then form the most beneficial one based on the generated knowledge. More precisely, each agent starts the optimization process by first determining all those coalitions including itself as a member. Afterwards, it calculates the corresponding coalition values in coordination with other included members, where the worth of a coalition is generally determined by the task which execution yields the highest possible benefit. As the number of coalition values is exponential in the number of agents, the algorithm restricts the size of a coalition through a parameter in order to reduce computational complexity. In the second stage, each agent then identifies the most beneficial of its calculated coalitions and shares the information with other agents. The exchanged data finally allow to form that coalition providing the maximum global benefit which is assigned to the task on which its coalition value is based. As this task may also have determined the values of other coalitions, these are finally recalculated if necessary before the next iteration is started.

With regard to our considered requirements, the proposed algorithm allows for a distributed and temporally flexible task-oriented formation of coalitions. However, because the heuristic optimizes global value and calculates all possible coalition values in the course of execution, it is associated with a high computational complexity of order $O(n^k \cdot |T|)$, with n being the number of agents, k being the maximum coalition size, and $|T|$ being the number of tasks. Moreover, it addresses no aspects related to the topics trust and value distribution.

Two further approaches for a decentralized formation of coalitions are presented by Sims, Goldman, and Lesser in [SGL03]. Although they consider the specific domain of distributed sensor networks, the authors specify their addressed question in terms of the CSG problem making their approaches applicable to other use cases as well. Basically, the proposed protocol classes are modified versions of the contract net protocol [Smi80, SD81], an approach for cooperative problem solving in distributed systems. The contract net protocol specifies a negotiation process in which a task manager broadcasts

the description of an open task to potential contractors which then provide bids for its fulfillment according to their given capabilities. After a specified deadline has passed, the manager awards a contract to the bidder with the most beneficial offer which is then obliged to fulfill the task. While this approach is also suitable for coalition formation in order to identify like-minded cooperation partners, it does not allow for an optimization of social welfare as specified by the CSG problem. Hence, the authors propose two similar protocols which are based on the concepts of local marginal utility and social marginal utility. In both cases, the negotiations take place between sector managers, i.e. coalition leaders which transfer members between their coalitions according to the given optimization goal. In the context of the local marginal utility based protocol, a manager broadcasts a request for cooperation to other agents which then provide an answer if a member transfer yields a positive local marginal utility, i.e. an increase in the value of their coalition. In contrast to the contract net protocol, a replying coalition leader is here not bound to its offer and thus able to answer an arbitrary number of requests in parallel. The initiating manager then selects the bid maximizing its local utility and sends a notification to the corresponding agent. The latter finally concludes the negotiations by confirming that cooperation which maximizes its own local utility as well. Contrary, in the context of the second protocol, agents make their choices based on the resulting social marginal utility, i.e. the sum of the local marginal utilities of both negotiation partners. As in case of a successful negotiation this value is always positive, it is guaranteed that the formation process results in an improved global utility even if the local utility of one cooperation partner is negative.
With regard to our specified requirements, the proposed protocols allow for a decentralized and temporally flexible formation of coalitions with the goal to maximize global value. As both approaches are based on local knowledge and communication takes place between managers only, they can also be expected to scale well with the number of agents. However, aspects related to trusted cooperations or the distribution of a gained surplus are not considered.
In [ASSR12], Anders et al. finally describe a set partitioning algorithm for distributed agents (SPADA) which can also be applied to the problem of coalition structure generation. The decentralized heuristic is based on a specific graph representation in which a coalition[3] of agents is modeled as a directed tree, where the root of the latter represents a coalition leader to which all other members are connected by means of labeled edges. The labels differentiate the members from acquainted agents which are also part of the tree but connected through unmarked edges. Given an arbitrary coalition structure as starting point (e.g. one in which all agents form a coalition on their own), in the course of execution each coalition leader then strives for an optimization of its coalition by iteratively assessing the reward of adding new acquaintances to and excluding current members from the tree. The calculation of the reward is based on a fitness function

[3] In fact, the authors use the word partition for a coalition and partitioning for the set of partitions resulting from the optimization process. As this terminology is inconsistent with the one typically used in the context of the CSG problem, we use the analogous terms coalition and coalition structure in order to avoid confusion.

which quantifies the value of a coalition based on local knowledge and has to be specified by the system designer according to the requirements of the given application. If the addition of an acquaintance is considered beneficial, the coalition leader sends an invitation to the agent which is free to accept it according to its own evaluation. If the invitee decides to join, it is transferred to the new coalition along with its acquaintances which thus become available as new potential members. Contrary, if the exclusion of a member is considered beneficial, the participant is informed about its removal after which it forms a new coalition on its own. Having examined a limited set of agents in a given optimization cycle, a coalition leader finally mixes the acquaintances of its members in order to promote the exchange of different agents between coalitions. If the specified optimization goal is then reached, it stops the process but becomes active again if one of its members receives a new invitation. Thus, the algorithm is generally not guaranteed to terminate.
With regard to our considered requirements, SPADA is an application-independent approach which allows for a decentralized formation of coalitions based on local knowledge. It can be expected to scale well with the number of agents as coalition leaders only consider a restricted number of them when optimizing their coalitions. Through an appropriate specification of the fitness function, the algorithm is also applicable to scenarios in which coalitions dynamically reorganize over time. However, although the heuristic is application-independent, it can not be applied to all optimization problems as the calculation rules for the rewards inherently prescribe the objective function. Moreover, the approach does not integrate aspects related to trusted cooperations or the distribution of a gained surplus.

1.2.4 Summary and Discussion

Table 1.1 gives an overview of the previously discussed approaches and provides an evaluation with regard to their suitability to fulfill the initially stated requirements. As depicted, most deficits exist with respect to a combined consideration of potentially conflicting optimization goals, an integration of trust-related aspects into the aggregation process, as well as a fair distribution of a resulting payoff.
The described concepts from the field of electric power provision generally reflect the trend towards more distributed and flexible aggregation methods which account for the current restructuring processes in the domain as discussed at the beginning of this chapter. While CVPPs and TVPPs are still based on the conventional paradigm of a centralized control, PowerMatcher and Lünsdorf's self-organization approach represent decentralized approaches providing an increased degree of scalability. However, despite their distributed nature, the latter still possess properties decreasing the system's robustness against failures: While PowerMatcher applies an internal hierarchical organization within a cluster, Lünsdorf's heuristic assigns an increased responsibility to agents with a high degree of centrality.
According to their intended purpose, the concepts from the domain of e-commerce enable a temporally flexible pooling of high numbers of customers for the joint purchase

approach	R_{DC}	R_S	R_{CO}	R_{TF}	R_{TC}	R_{PD}	R_{TA}
CVPP/TVPP							•
PowerMatcher	•	•					
virtual devices	•	•		•			•
post-/pre-negotiation protocol		•		•			
GroupBuyAuction		•		•		○	
coalition formation via bilateral Shapley value	•	○		•		○	
task allocation via coalition formation	•			•			
local/social marginal utility based protocol	•	•		•			
SPADA	•	•		•			

• requirement fully addressed
○ requirement partly addressed

Table 1.1: Related work from the domains of electric power provision, e-commerce, and DAI.

of selected products. Yet, both discussed approaches only allow for a centralized aggregation, where scalability is either achieved by applying an efficient heuristic (GroupBuyAuction) or performing no optimization at all (post-/pre-negotiation protocol). With regard to a division of the resulting profit, GroupBuyAuction integrates a stable but not fair distribution scheme.
Finally, the agent-based concepts from the domain of DAI allow for both a goal-oriented and completely distributed formation of coalitions. However, the discussed approaches demonstrate that a decentralized control does not automatically result in a high degree of scalability. If the decision-making process of the agents is based on a global information model (as it holds for the first two of the described concepts), the computational and communication cost become unmanageable in case of large numbers of actors, thus outweighing the advantages introduced by a distributed design.
From the previous descriptions we can finally conclude that there is yet no approach meeting all of our specified requirements. However, the discussed solutions still provide several valuable features and insights for the development of a new aggregation method:

1. Decentralized interaction protocols as implemented by the utility based protocols and SPADA are suitable for building coalitions with flat organizational structures. Moreover, if the decision-making process of the agents is based on local information only, they can also be efficiently applied to large-scale scenarios comprising high numbers of actors.

2. Topology-based neighborhoods as used in Lünsdorf's self-organization approach provide a reasonable means in order to integrate grid-related aspects into the formation process and restrict a pool to a specific grid section if required by the pursued goal.

3. A payoff distribution can be decoupled from the actual aggregation process if the individual share of an agent is no decision criteria for its participation in a coalition. As can be seen from the example of GroupBuyAuction, the division can then be performed after coalition formation has finished.

1.3 Objectives

Having motivated the need for a new aggregation method and outlined the advantages and drawbacks of present approaches with regard to the corresponding requirements, we are finally ready to detail the objectives of this thesis. These are as follows:

Main Goal

Goal of this thesis is the design of an aggregation method which allows computerized, interconnected energy units of type producer, consumer, and storage to form pools in a self-organized and fully decentralized fashion. As general setting, we consider electricity markets in which participants trade power products either bilaterally or via a mediating actor like an exchange. If the properties of a product exceed the operational capabilities of a unit, it strives for a cooperation with other trustworthy participants in order to achieve a joint fulfillment. The trade of a fulfilled product finally yields a profit which is to be divided among the cooperation partners in a way which is fair and generally incentivizes participants to declare their contributions to a pool correctly and truthfully in order to promote contract compliance. With regard to its optimization goals, the new method is expected to take both the preferences of the individual actors and those of the global supply system into account. On the one hand, it shall allow for an optimization of local profits through the identification of a beneficial combination of products exploiting the operational capabilities of a unit. On the other hand, it shall also enable an optimization of social welfare by promoting the formation of pools improving the utility of the system as a whole. Moreover, the approach is intended to be temporally flexible in the sense that the organizational binding of a participant to a pool is restricted to the fulfillment of a single product only. Finally, it shall be applicable to any market type, scale well with the number of units, and allow for the trade of localized power products through the integration of appropriate topological information.

Although a product-oriented formation of pools might seem quite restrictive at first sight because actors are in this case only able to cooperate if they strive for the same contractual conditions, we expect our new approach to allow participants to flexibly adopt the conditions of potential cooperation partners in the course of a formation process. Moreover, actors shall be able to identify an alternative combination of products if the formation process for a given product was unsuccessful.
By developing an aggregation method providing the above features, we strive for an approach which fulfills the previously defined requirements and thus solves the identified problems of present concepts. Because of its desirable properties, we expect the method to yield high quality solutions for the given optimization problems even for large-scale

scenarios comprising high numbers of actors. We capture these expectations in the form of the following hypotheses which we finally use as basis for evaluating the approach in Chapter 5.

Hypothesis 1.1: Local Performance

Through integration of an appropriate optimization algorithm, a distributed aggregation method allows actors to successfully optimize their expected utility and approximate the optimal solution of the corresponding local optimization problem to a high degree.

The first hypothesis captures our assumption that through decentralization actors are able to optimize their expected individual profits resulting from the trade of jointly fulfilled products. As described later in Section 4.1.2, these benefits generally depend on the attributes of the products as well as the operational potentials of the energy units. Besides the above property, we also assume that our new approach performs well on global system level.

Hypothesis 1.2: Global Performance

A distributed and temporally flexible aggregation method enables a high rate of actors to fulfill their power products and allows to approximate the optimal solution of the corresponding global optimization problem to a high degree.

I.e., as each actor strives for the fulfillment of its desired products, we expect the method to enable a high percentage of participants to achieve this objective and thus to optimize social welfare. Finally, we assume that the approach can be efficiently applied according to its intended purpose, where a scenario may potentially comprise a large number of actors.

Hypothesis 1.3: Efficiency

A distributed aggregation method is efficiently applicable in terms of the computational cost, communication cost, and runtime which it requires to provide a solution.

I.e., in order to assess the efficiency of our new approach, we consider the computational cost caused by the global optimization process, the communication cost caused by the actors through the coordination of their actions, as well as the method's total runtime until termination. Specific measures for quantifying these and the above stated performance-related criteria are defined in the context of evaluation in Chapter 5.

1.4 Research Methodology

According to our objectives as stated in the previous section, in the following chapters we design and evaluate a new aggregation method for the decentralized pooling of energy units. In order to structure the related activities, we apply a research methodology

which is basically an integrated approach combining two existing paradigms. As depicted in more detail in Figure 1.1, the latter are one the one hand given by the design science research guidelines (DSRGs) [HMPR04] which we use as general requirements for our research and on the other hand by the Smart Grid Algorithm Engineering (SGAE) process model [NTS14] which we apply for the organization of the performed activities. Both paradigms complement each other in the sense that, if extended by a single step, the process phases defined by the SGAE approach allow for a research being in line with the guidelines. In the following, we detail the applied methodology and then outline our conducted research activities.
The design science research guidelines were originally proposed by Hevner et al. in order to promote the quality of research in the context of information systems. In contrast to natural science, which purpose is the development and justification of scientific claims like theories or laws, design science strives for the creation and evaluation of artifacts which can either be constructs (vocabulary), models, methods, or instantiations [MS95]. In this sense, we can view our work as design science research as we seek to create and finally evaluate a new aggregation method for the pooling of energy units. Although in their publication Henver et al. refer to the discipline of information systems, their view on design science is generally in line with the afore mentioned understanding making the proposed guidelines transferable to other areas as well. The different principles can thus be summarized as follows [HMPR04]:

- *Guideline 1 (Design as an Artifact)* requires that design science research has to create an artifact, i.e. a construct, a model, a method, or an instantiation. This artifact has to be appropriately described in order to allow for a later implementation or application.

- *Guideline 2 (Problem Relevance)* requires that the built artifact must be valuable in the sense that it provides a solution to a relevant problem.

- *Guideline 3 (Design Evaluation)* requires that the utility, quality, and efficiency of the artifact must be validated through adequate evaluation methods like experimental, analytical, or descriptive approaches. These are typically retrieved from a knowledge base, i.e. a central repository being composed of existing foundations and methodologies. The choice of appropriate methods generally depends on the artifact as well as the metrics used for evaluation.

- *Guideline 4 (Research Contributions)* requires that design science research has to provide one or more verifiable contributions which add to the existing contents of the knowledge base. Besides the artifact itself, these can also comprise other foundations or methodologies which result as additional products from the work process.

- *Guideline 5 (Research Rigor)* requires that design science research has to create and evaluate the artifact by means of rigorous methods. This is achieved through the application of appropriate techniques and an effective utilization of the knowledge base.

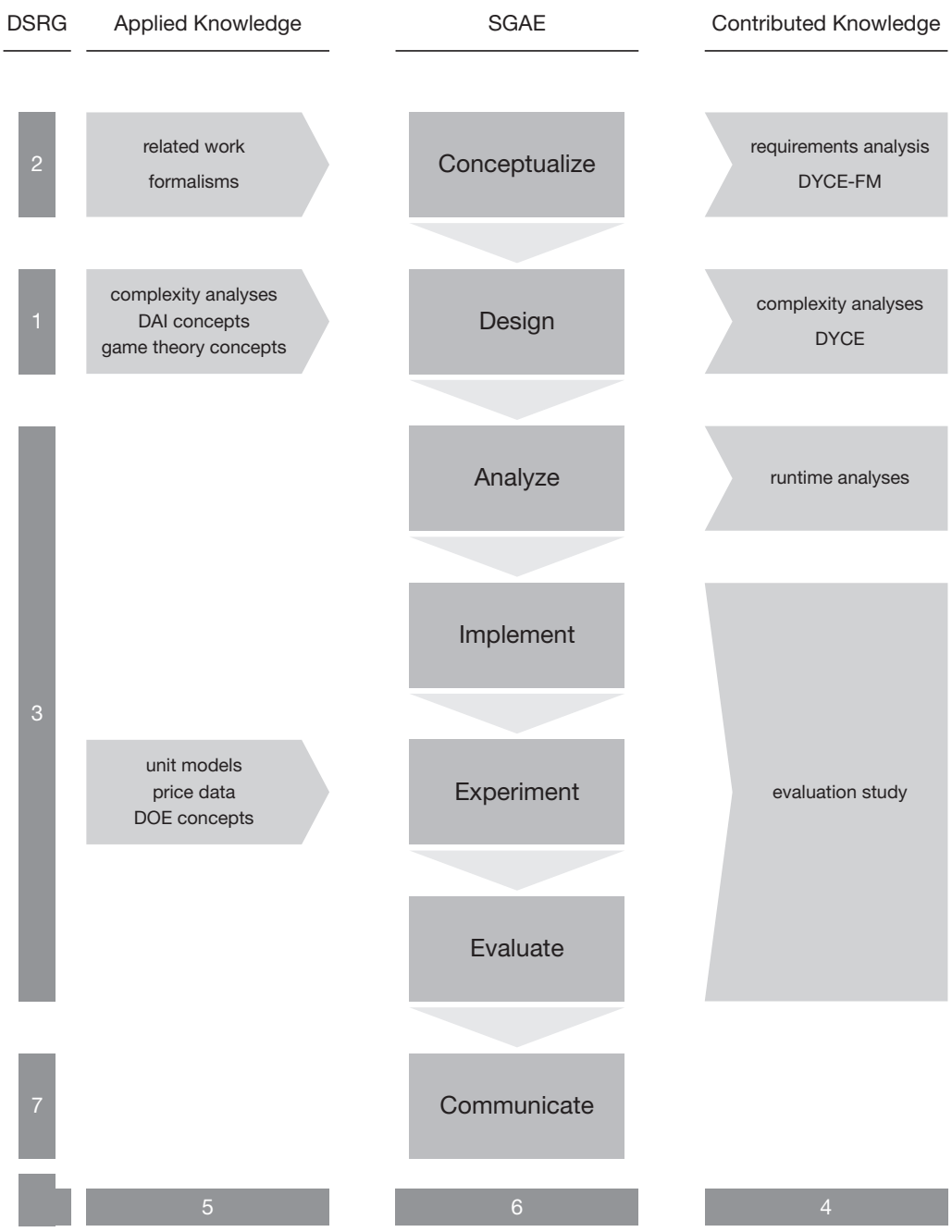


Figure 1.1: Applied research methodology.

- *Guideline 6 (Design as a Search Process)* requires that the artifact has to be designed by means of a search process which uses available means to achieve the desired ends while satisfying the laws existing in the problem environment. If it is impossible to find the optimal design for an artifact, a heuristic search strategy has to be applied which allows to achieve a satisfying solution.

- *Guideline 7 (Communication of Research)* finally requires that design science research has to be presented to both technology-oriented and management-oriented audiences. In either case, it must be communicated in a way satisfying the needs of the addressed group.

As can be seen from the above descriptions, the proposed guidelines specify requirements for the performance and presentation of design science research. However, they do not define a specific workflow, i.e. a sequence in which the related activities have to be conducted. As second paradigm, we thus apply the Smart Grid Algorithm Engineering approach which specifies an iterative process model for the development of algorithms in the context of power systems. As throughout this thesis we only conduct a single iteration of the defined development cycle, we run through a sequence of work steps as depicted in Figure 1.1. As already mentioned above, in order to address all of the seven research guidelines, it was required to add an additional step at the end of the sequence (Communicate) which is not part of the original model. Besides the different work steps, Figure 1.1 also shows the associated inputs and outputs in terms of the existing knowledge which we apply in the context our research as well as the new knowledge which we finally provide in the form of the resulting contributions. Referring to the depicted methodology, the different research activities performed throughout this thesis can be summarized as follows (for a general description of the SGAE phases see [NTS14]):

Conceptualize As first work step, we analyze and specify the addressed problem in-depth. Throughout the previous sections, we already began with this task by motivating the need for a new aggregation method and identifying related requirements based on shortcomings of present pooling concepts. Moreover, we detailed our goal to provide a new solution and formulated hypotheses in order to capture our expectations with regard its performance and efficiency. In the following chapters, we complete the work step by first discussing the fundamental topics which are relevant for the design and experimental evaluation of the intended approach. Moreover, we specify DYCE-FM, a formal model comprising concise definitions of all concepts being crucial for the development process. As we expect our new aggregation method to be universally applicable to a wide range of use cases, the definitions abstract from specific details like the type of a market or the technology of an energy unit. According to our goal to design a decentralized system, the specifications partly draw on formalisms from the domains of distributed artificial intelligence and game theory.

Design Based on the formal model created in the previous step, we next develop a new method for the decentralized aggregation of energy units referred to as DYCE[4] – DYnamic Coalition formation in Electricity markets. To this end, we first partition the considered problem into four subproblems and analyze the related computational complexities partly drawing on existing proofs where available. Based on the results, we then develop an appropriate solution concept for each subproblem which forms an integrated part of the sought overall solution. Drawing on concepts from the domains of DAI and game theory, the new aggregation method is generally designed as a distributed approach in which each energy unit is controlled by an autonomous intelligent agent acting as its representative in the market. Based on individually specified product portfolios exploiting the operational capabilities of their units, in the course of execution the agents then coordinate their actions in order to form coalitions and corporately fulfill their pursued products. If a formation effort was successful, a coalition strives for a trade of its fulfilled product providing a profit which is distributed among the members in a fair way. Throughout the whole process, the agents maintain an abstract model of the power grid which allows them to assess the physical distance to other units in the grid. This way, they are able to dynamically form neighborhoods of nearest neighbors within which the actual formation of coalitions takes place. Neighborhoods thus allow to restrict the distribution of coalition members over the grid and to handle the computational and communication cost associated with the formation process.

Analyze In order to assess the computational complexity which is associated with the previously developed design, as third step we examine the running times of the integrated optimization algorithms in terms of the O-notation. With regard to a practical application, these analyses allow for an estimation of the requirements which have to be fulfilled by the hardware components running the agents.

Implement As prerequisite for an experimental evaluation, we next implement the developed agent-based design and tie it to a simulation system. The latter enables an investigation of different scenarios and parameterizations based on discrete-event simulations.

Experiment In order to examine the initially stated hypotheses, as fifth step we specify a set of research questions with regard to the performance and efficiency of the developed approach. For each question, we then conduct an experiment which comprises a set of simulations being executed on the basis of an appropriately designed scenario. The results are used in the next work step in order to provide answers to the specified questions. The investigations are primarily based on techniques from the domain of Design of Experiments (DOE) which allow to maximize the gained information and minimize the experimental efforts.

[4] pronounced as [daɪs]

Evaluate Given the output data obtained from the conducted experiments, we are next able to evaluate the developed aggregation method. To this end, we answer the previously specified research questions based on defined quality measures and draw conclusions with regard to the performance and efficiency of the approach which eventually allows us to support or reject the initially stated hypotheses.

Communicate As last step, we finally publish the results of the conducted research through publication of this thesis.

As shown in Figure 1.1, the above work process is generally in line with the previously discussed design science research guidelines. I.e., we address guideline 1 (Design as an Artifact) by developing a new aggregation method in step 2; guideline 2 (Problem Relevance) by motivating the need for the method in step 1; guideline 3 (Design Evaluation) by analyzing its computational complexity in step 3 and further characteristics in step 4-6; guideline 4 (Research Contributions) by providing new knowledge in the form of several results; guideline 5 (Research Rigor) by drawing on existing foundations and methodologies where appropriate; guideline 6 (Design as a Search Process) by applying the SGAE model for the organization of our research process; and guideline 7 (Communication of Research) by finally publishing the results in step 7. This way, we apply a research methodology which provides a well-founded basis for the achievement of our goals.

1.5 Thesis Structure

The structure of this thesis basically follows the previously outlined research process, where descriptions may span different steps if it improves readability. For instance, we discuss the running time of the proposed optimization algorithms directly after their specification rather than in a dedicated chapter.
Next, Chapter 2 covers the topics which make up the background of this thesis by giving an overview of the deregulated electricity industry and providing an introduction to the fields of distributed artificial intelligence and Design of Experiments. Afterwards, Chapter 3 discusses the problem of coalition formation in electricity markets in-depth and specifies the above mentioned formal model which forms the basis for the following design and evaluation of our intended aggregation method. The latter is then developed in Chapter 4 which also comprises the complexity analyses of the considered subproblems as well as the runtime analyses of the included optimization algorithms. Afterwards, Chapter 5 describes the experimental evaluation of the approach and discusses the results in view of the initially stated hypotheses. Chapter 6 finally concludes the thesis by reviewing the results and proposing topics for future research.

2 Background

Having motivated the need for a new aggregation method, in the following sections we provide an introduction to the fundamental topics which underlie the contents of this thesis. In Section 2.1 we start by describing the basic structure of the liberalized electricity industry and discussing the different market types which can be used for the trade of electrical energy. In Section 2.2 we then cover the IT-related background by detailing the concept of an intelligent agent and the inner workings of multiagent systems. As last topic, in Section 2.3 we provide an introduction to Design of Experiments, a comprehensive and efficient approach for experimentation which we extensively apply in the context of evaluation in Chapter 5. Section 2.4 finally summarizes the most important aspects and discusses them in view of our objectives.

2.1 Electricity Industry

In the following sections, we provide a brief overview of the deregulated electricity industry and the German electricity market. First, Section 2.1.1 describes the industry's basic structure along with the physical and economic processes which connect the associated actors. Afterwards, Section 2.1.2 provides a closer look at electricity markets and the different market types which can be applied in order to achieve an economic equilibrium between supply and demand. The descriptions are mainly based on [SPH12, vWS05, Sto02].

2.1.1 Structure and Processes

In contrast to other economic sectors, the liberalized electricity industry is particularly characterized by the properties of its underlying commodity. As to date electrical energy is still not storable in significant amounts and there is thus no possibility to compensate an underproduction by means of a prior overproduction, supply and demand have to be matched at each point in time. Moreover, in contrast to other commodities, electricity can only be exchanged between producers and consumers which are connected to the same supply network. As a consequence, if two electricity markets relate to distinct, unconnected networks, price differences do not induce a higher production in the higher priced market. Finally, because of its physical characteristics, electricity can hardly be substituted by other energy sources which results in a strong dependency on the commodity and a low price elasticity of demand [SPH12].
The specific properties of electrical energy have considerable consequences on the way how it is traded and physically supplied. As already pointed out in Chapter 1, the organizational structure of the electricity industry can generally be distinguished into an economic and a physical level. Both are interrelated in the sense that electrical energy is traded on the economic level via markets and then delivered on the physical level through a technical supply network. Figure 2.1 shows this basic organization along with the different roles and processes which exist within the electricity industry. Generally,

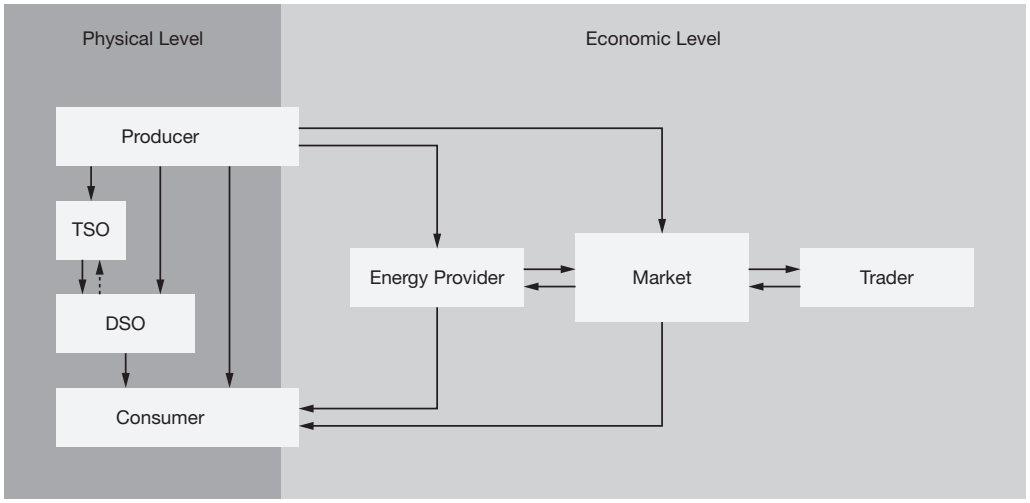


Figure 2.1: Roles and processes within the electricity industry (adapted from [vWS05]).

a single actor can take on more than one of the depicted roles, like a household which consumes electricity but also operates a local CHP plant for production. As shown on the left hand side of the figure, in the course of the physical delivery process producers convert primary energy into electrical energy which is then fed into the transmission and distribution grid being managed by transmission system operators (TSOs) and distribution system operators (DSOs), respectively. More precisely, in case of large-scale generators like nuclear or coal-fired plants, the electricity is fed into the transmission network and transported to the distribution network which finally supplies it to the consumers which are connected to the medium and low voltage grid. Contrary, if produced by small- and medium-sized units like solar power or wind energy plants, the electricity is directly fed into the distribution grid, where situations can arise in which the supply exceeds the demand and the energy has to be fed back into the transmission network. Finally, the electricity can also be produced directly by units on-site (like in case of domestic solar power plants) and consumed locally without the use of the public supply network.

The physical delivery of electricity is generally based on contracts which are concluded on the economic level of the system. As shown on the right hand side of Figure 2.1, supply and demand are matched via dedicated markets which allow participants to trade power products according to their individual interests. A product specifies the amount of electricity which is offered or requested at a certain price with regard to a future time interval. As described in more detail in the next section, today's electricity markets typically comprise different submarkets (like a day-ahead or an intraday market) which allow actors to trade products with regard to different time scales. As small-scale consumers like households do not directly participate in trades at wholesale markets, they are typically supplied by energy providers which cover their demand either through own

power plants or by means of appropriate purchases. As a result, providers can also resell excess amounts of electricity which they bought in order to ensure a secure supply. As further market participants, traders also buy and sell electrical energy but, in contrast to providers, do not supply any customers.
Generally, the above trading processes require producers, consumers, and energy providers to make predictions with regard to their expected production and demand which form the basis for their sales and purchases. Until a specific deadline before physical fulfillment, deviations from these predictions can compensated through trades, while in the course of provision DSOs and TSOs are responsible for keeping the balance between supply and demand. To this end, they use balancing power which is typically traded at dedicated markets (not shown in Figure 2.1) [SPH12, Sto02].

2.1.2 Electricity Markets

As already outlined in the previous section, a deregulated electricity market typically consists of different submarkets which allow participants to trade electricity and related commodities with regard to different time scales. These can include long-term oriented forward and derivatives markets, short-term oriented day-ahead and intraday markets, as well as markets for the trade of associated commodities like control reserve or transmission rights. The submarkets of an entire market can be connected through implicit or explicit linkages, where the former result from unenforced market mechanisms like arbitrage, while the latter are deliberately created through market rules which may for instance require a purchase of transmission rights before a trade at the forward market. Moreover, the submarkets may be of a different type which determines how the trades between participants are organized. As shown in Figure 2.2, these can generally be arranged in a bilateral or a mediated fashion. While in bilateral markets buyers and sellers trade commodities directly (which are thus also referred to as over-the-counter markets), in mediated markets contracts are made with a mediating third party which acts as direct trading partner for both sides. Bilateral markets can be further distinguished into *direct search markets* in which participants have to find each other on their own, *bulletin board markets* in which the search is facilitated through a central medium like a web page, and *brokered markets* in which the search is performed by specialized agents which arrange the trades between buyers and sellers. Contrary, mediated markets can be further distinguished into *dealer markets* in which participants conclude contracts with dealers buying and selling traded commodities on their own accounts, *exchanges* which match supply and demand by means of a central auction, and *pools* which also use a central auction for the matching but further integrate additional information like the startup costs of generators or the costs of potential congestions in the network. While bilateral markets generally offer greater flexibilities as contract conditions can be specified individually, they are also associated with higher costs as the specification process requires higher efforts and the trading partners have to assess the creditworthiness of the counterparty on their own. Contrary, mediated markets provide less flexibility with regard to contract design but a higher degree of organization and centralization. For instance,

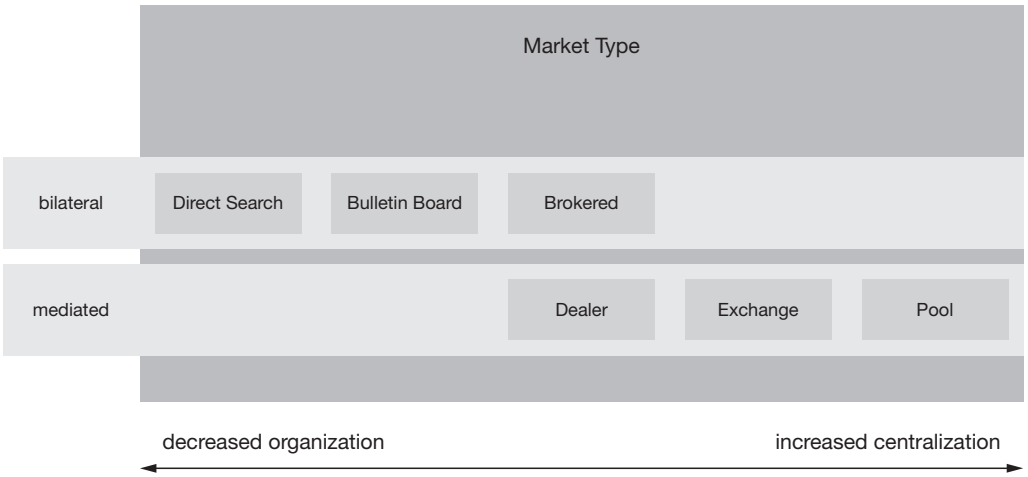


Figure 2.2: Types of electricity markets (adapted from [Sto02]).

exchanges allow for a trade of standardized products and the assessment of a unique market clearing price which is typically used as reference price for trades in bilateral markets [Sto02, BKM14, OGZ08].
To give an example, Figure 2.3 depicts the structure of the German electricity market. As shown, the included submarkets comprise a bilateral forward market and a mediated derivatives market for the trade of long-term contracts and options. The spot markets are given by a day-ahead and an intraday market which are both integrated into the European power exchange EPEX SPOT which also operates the spot markets of France, Austria, and Switzerland. Finally, there is a dedicated market for control reserve which is operated by the TSOs for the procurement of primary, secondary, and tertiary control reserve in order to meet their responsibility to ensure a safe operation of the power grid [SPH12].

2.2 Distributed Artificial Intelligence

As described in Section 1.3, goal of this thesis is the development of a new aggregation method which partly draws on concepts from the domain of distributed artificial intelligence. In the following, we make this objective clearer by providing an introduction to the field. In Section 2.2.1, we start by discussing the properties of an intelligent agent and characterizing the environments in which it can potentially act. In Section 2.2.2, we then provide an overview of multiagent systems and consider aspects of communication, coordination, and organization. The descriptions mainly draw on [RN10, WJ95, HS01, Syc98, HL04].

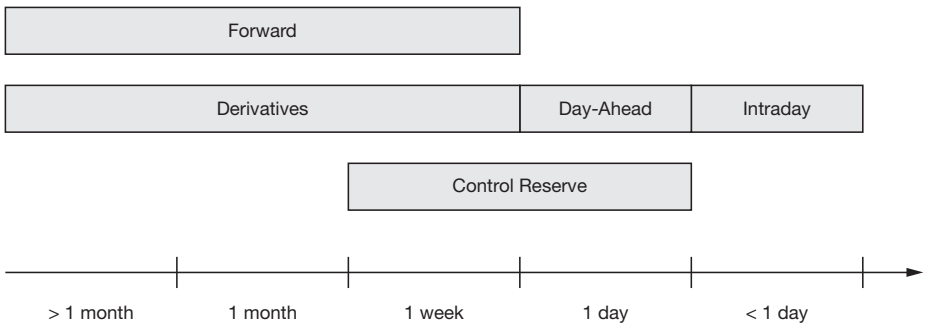


Figure 2.3: Structure of the German electricity market (adapted from [SPH12]).

2.2.1 Intelligent Agents

As can be seen from the discussion in [FG97], today there is still no consensus with regard to a commonly accepted definition of the term agent. The varying views particularly result from the requirements which are associated with the different application domains. One of the most general definitions is given by Russell and Norvig in [RN10]:

> "*An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.*"

This understanding is visualized in more detail in Figure 2.4. As shown, the sensed percepts are mapped by an agent program to actions which determine the agent's behavior. This way, it is able to respond to changes in the environment in an appropriate fashion. Generally, the agent program can be of different complexity according to the agent's purpose. For instance, it can be based on simple hard-coded condition-action rules which directly map percepts to actions or rely on more sophisticated search and planning methods enabling the agent to reason about its past and future behavior. The implemented decision making process generally takes a single percept or a longer percept sequence (i.e. a history of environmental states) into account. The resulting behavior can be rated by means of a performance measure which quantifies the agent's success in fulfilling its executed task.
Clearly, the above definition captures a very broad understanding of agency. For instance, it allows to view industrial robots or just simple Java methods as agents. However, there are obviously differences in the capabilities of these two agents with respect to the autonomy and flexibility to execute actions. In other words, they can be expected to exhibit a different degree of intelligent behavior. This aspect is reflected in the following more specific definition of a rational agent [RN10]:

> "*For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has.*"

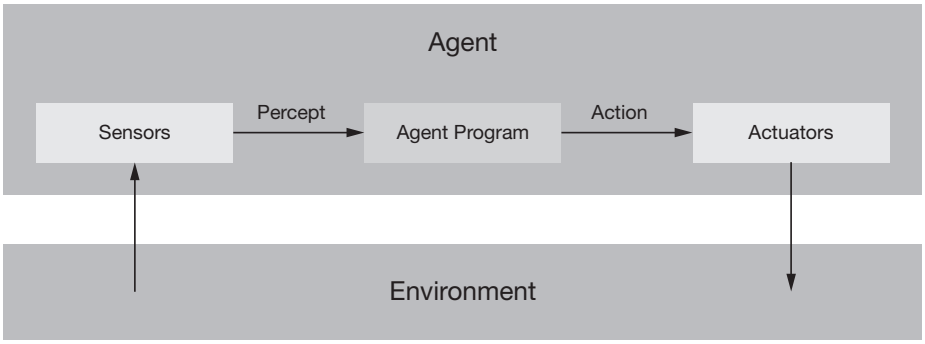


Figure 2.4: Concept of an agent (adapted from [RN10]).

A rational agent thus acts intelligently by always choosing an action which maximizes its success based on all available information. The extent to which this choice is based on its own experiences (and not on its hard-coded built-in knowledge) determines its degree of autonomy.
While this second definition integrates a reasonable notion of intelligent behavior, it only implicitly addresses situations in which multiple agents interact in order to fulfill their goals. In such cases, the rational choice of an action is particularly determined by the ability to appropriately respond to messages received from other actors. In [WJ95], Wooldridge and Jennings emphasize this aspect by defining an agent as a computer system possessing the following properties:

- *autonomy*, i.e. the capability to operate without external intervention and to control the own behavior and internal state,
- *social ability*, i.e. the capability to interact with other actors based on an agent communication language (ACL),
- *reactivity*, i.e. the ability to perceive the environment and respond to occurring changes in a timely fashion, and
- *pro-activeness*, i.e. the capability to act goal-directed and take the initiative in order to fulfill the pursued goal.

This understanding is generally in line with the definition of a rational agent but underlines its possible application in distributed systems in which multiple actors interact. As described in more detail below, social behavior is then a pivotal property in order to achieve global coherence through coordination.
Depending on its intended purpose, an agent can be situated in a broad range of different environments. The characteristics of a given setting have a decisive impact on the complexity of its design as they require different capabilities with respect to the perception and prediction of the world's state. In this regard, an environment can generally be

- *fully observable* or *partially observable*, where it is fully observable if it is completely perceivable by an agent,
- *deterministic* or *stochastic*, where it is deterministic if its next state solely depends on the current state and an agent's chosen action,
- *episodic* or *sequential*, where it is episodic if in a sequence of disjunct episodes in which an agent perceives and acts merely once the agent's success always depends on the current episode only,
- *static* or *dynamic*, where it is static if no changes occur while an agent is reasoning,
- *discrete* or *continuous*, where it is discrete if states, percepts, actions, and time can be handled in a discrete fashion,
- *known* or *unknown*, where it is known if an agent possesses complete knowledge about the consequences of its actions, and
- a *single agent* or *multiagent* environment, where it is a single agent environment if it comprises one agent only [RN10].

If an environment is only partially observable, an agent has to maintain an internal model in order to keep track of the elements which are out of its range of perception but relevant for decision making. From its own viewpoint, a complex deterministic environment may thus also appear to be stochastic even if the opposite is the case. Deterministic environments are generally easier to handle because there is no uncertainty about the result of a chosen action. Sophisticated capabilities are required in sequential and dynamic environments in which actions have an impact on the future performance of an agent and it has to be steadily aware of potential changes. The same holds for continuous and unknown environments in which an agent is confronted with continuous properties and has to learn which consequences result from its actions. Finally, if an agent is situated in a multiagent environment, its performance is also influenced by other agents which requires communication and coordination among the actors as described in more detail next.

2.2.2 Multiagent Systems

In recent years, multiagent systems have gained a lot of attention as paradigm for building large-scale distributed systems being composed of potentially heterogeneous components which can enter or leave the system over time. The basic structure of a multiagent system is depicted in Figure 2.5. As shown, the involved agents are able to coordinate their behavior through structured interactions in order to fulfill their pursued goals in a coherent fashion. To this end, they may also form organizational structures which potentially evolve over time according to their intended purpose. The key properties of a multiagent system are that

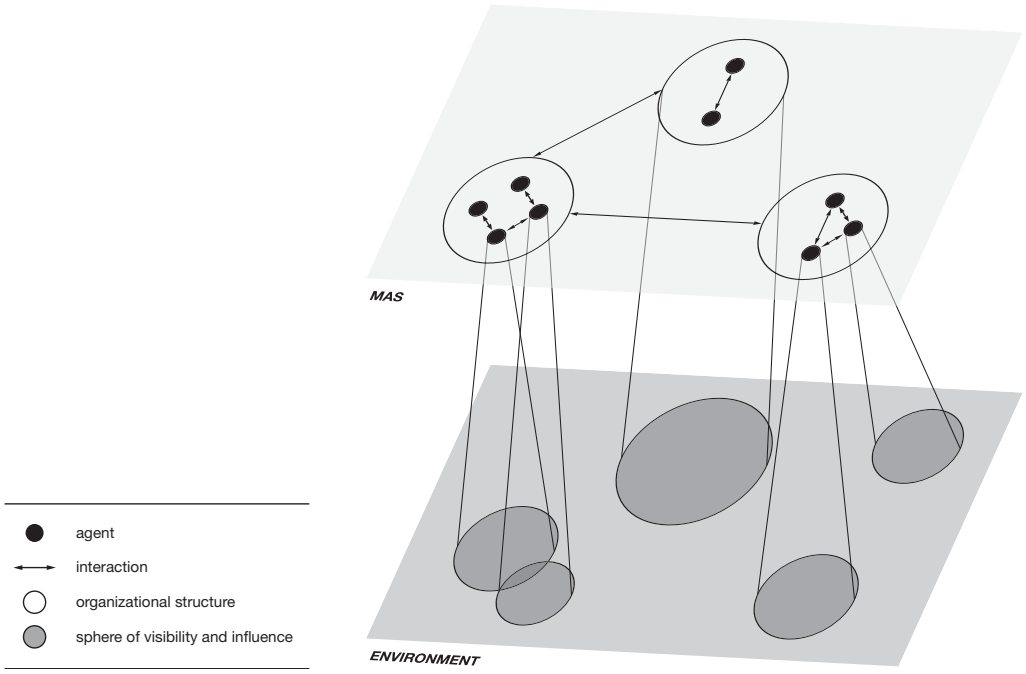


Figure 2.5: Basic structure of a multiagent system (adapted from [Jen01]).

- each agent has only incomplete information or capabilities in order to fulfill its goal,
- system control and data are decentralized, and
- computations are performed in an asynchronous fashion [Syc98].

Although agents are often simply not able to execute a specific task on their own, they can also coordinate their actions in order to achieve efficiency gains. For instance, in an electronic market, a buyer agent may indeed be able to purchase single products on its own but strive for a cooperation in order to obtain volume discounts through the trade of wholesale lots (cf. Section 1.2.2).
Because of their distributed nature, multiagent systems provide several advantages over centralized architectures. First, they allow for an increased efficiency and flexibility as calculations are performed in a parallel fashion and capabilities of different agents can be dynamically combined according to a given task. Moreover, failures of single agents can be compensated by other agents with similar capabilities resulting in an increased reliability at runtime. Finally, multiagent systems can be developed, maintained, and extended more efficiently and flexibly through a cost-effective component-based design [Wei01, Syc98].

2.2.2.1 Communication

In order to coordinate their actions, the agents of a multiagent system communicate based on a dedicated agent communication language. As a specific form of communication protocol, an ACL allows for a standardized specification and transfer of single messages and thus for an information exchange between two communication partners. Today, the most established languages are given by the *Knowledge Query and Manipulation Language* (KQML) and the more recent FIPA ACL which was developed by the *Foundation for Intelligent Physical Agents* (FIPA) as part of their standardization efforts in the context of agent-based systems [LFP99]. Both represent high-level languages in the sense that they specify the properties of a message independent from its content and the protocol applied for its transfer. Moreover, both define the type of a message in terms of speech act theory which considers utterances in natural language as actions by which a speaker influences its environment. A speech act generally consists of its physical utterance (referred to as *locution*), its intended meaning (*illocution*), and possibly the resulting action (*perlocution*) [HS01]. For instance, according to speech act theory, the utterance "Please switch off the light." is further characterized by its meaning as a request and – if the request is met – the effect that the light is turned off. Both KQML and FIPA ACL make use of this notion by defining the type of a message in terms of its illocutionary force. In case of FIPA ACL, the corresponding message parameter is referred to as *Communicative Act* which can take on values like *inform*, *request*, *agree*, or *cancel*. Besides an informal description of all possible alternatives which can be specified by a sender, the FIPA communicative act library also provides formal definitions of their underling semantics including the *feasibility preconditions* which must hold before a performance and the *rational effect* which is expected by a sender as a result [Fou02b]. This way, the meaning of a message can be unambiguously defined so that communicating agents have a common understanding of the transmitted content. To illustrate, Figure 2.6 gives an example of a FIPA ACL message which might be sent by an electricity consumer to a producer as a proposal for a trade. As shown, besides the identifiers of both communication partners, the message parameters also specify the actual content as well as the language in which it is expressed. A complete listing of all possible parameters can be found in [Fou02a].

2.2.2.2 Coordination and Organization

The ability to communicate enables agents to interact and coordinate their behavior. Coordination is generally defined as a process in which a set of agents acts in a coherent fashion, thus avoiding undesirable situations like livelocks, deadlocks, or resource contentions. As the agents of a multiagent system have joint or interdependent goals and only possess limited capabilities or information in order to fulfill the very same, the application of an appropriate coordination technique is a pivotal requirement for their success. Moreover, there may be global constraints like deadlines for common tasks which require interactions [NLJ96].

```
(propose
     :sender        consumer_ID
     :receiver      producer_ID
     :language      XML
     :content       (<product>
                         <amount>
                              <value> 100 </value>
                              <unit> kWh </unit>
                         </amount>
                         <price>
                              <value> 0.1 </value>
                              <unit> EUR/kWh </unit>
                         </price>
                    </product>)
)
```

Figure 2.6: Exemplary FIPA ACL message.

Generally, coordination techniques can be distinguished into cooperative and competitive approaches which are applicable in systems of non-antagonistic and self-interested agents, respectively. As shown in Figure 2.7, these can be further classified into the following subcategories [NLJ96, HS01]:

Organizational structuring An organizational structure defines the roles, authorities, and relationships of the agents in a multiagent system (cf. Figure 2.5). This way, it constrains their actions to predefined rules and allows for a coordinated behavior. Generally, an organizational structure can be temporally fixed or evolve over time according to the course of the interactions. For instance, the agents of a MAS could be arranged into a fixed hierarchy which is predefined at design time. Alternatively, they could also organize themselves into dynamic coalitions which adapt to environmental changes at runtime. The specific characteristics of different organizational structures are discussed in more detail below.

Contracting Another cooperative coordination technique is given by the contract net protocol, a well-established approach for distributed problem solving which was first proposed by Smith and Davis [Smi80, SD81] and later standardized by the Foundation for Intelligent Physical Agents with minor modifications [Fou02c]. The basic idea of distributed problem solving is to decompose and solve a given problem in a decentralized fashion and finally integrate the generated partial solutions into the sought overall solution. According to this approach, the contract net protocol specifies a distributed bidding process in which a manager announces an open task (i.e. a subproblem) to contractors which then submit bids for its execution based on their capabilities. Given the offers received as a reply, the manager finally awards a contract to the most appropriate bidder. By taking on the role of a manager, a contractor is generally able to further decompose an awarded task and initiate the same bidding process which consequently results in a hierarchical decomposition structure.

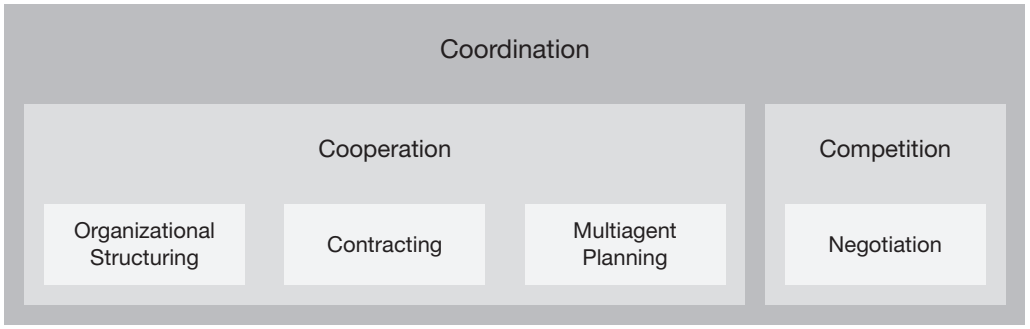


Figure 2.7: Classification of coordination techniques (adapted from [HS01]).

Multiagent planning In the context of multiagent planning, agents coordinate their behavior cooperatively through plans which specify their intended future actions. This way, they are able to identify potential conflicts and resolve them through appropriate modifications. Generally, the coordination can be realized in a centralized or a distributed fashion, where in the former case a single agent is responsible for analyzing the individual plans of the whole community and integrating them into a consistent overall plan. Contrary, in the context of distributed multiagent planning, agents steadily exchange their individual plans in a decentralized fashion which allows them to resolve potential conflicts without a central coordinator. Although this technique is associated with a higher complexity, it preserves the benefits of a distributed system.

Negotiation In the course of a negotiation, two or more self-interested agents strive for a joint agreement. Because their goals are potentially contradicting, they exchange their individual positions and make proposals for a solution based on their applied strategies. By making concessions, they try to solve conflicts and reach an agreement which is acceptable for all involved negotiation partners.

Generally, the above categories are not strictly separable in the sense that many coordination techniques are based on several of the described classes. For instance, the contract net protocol can be applied in order to identify appropriate cooperation partners for beneficial organizational structures (cf. Section 1.2.3). Moreover, the term negotiation is frequently used in the context of cooperative methods in order to describe an interaction process among agents.
Besides enabling coordinated behavior, an organizational structure can further allow agents to increase their individual power, achieve more challenging goals, or handle potential uncertainties. To date, various approaches have been proposed which properties make them applicable to different use cases. While the following list provides a description of typical organizational structures, a more comprehensive overview can be found in [HL04].

Hierarchies In a hierarchy, agents are arranged into a tree with two or more levels which branches define their communication links. Typically, the levels reflect the authority

structure of the organization in the sense that higher located members control the ones below them. However, in order to improve efficiency and prevent an overload of the root agent, increased responsibilities can also be allocated to other nodes of the tree. Hierarchies allow to control the data flow between agents and manage communication cost. However, as main drawback they generally lack in robustness as a failure of single agents can have severe consequences on the functionality of other members.

Holarchies A holarchy can be considered as a special form of hierarchy in which all tree nodes represent partially-autonomous entities termed *holons*. As a functionally integrated unit, a holon consists of all subordinate holons and is in turn part of all superordinate holons in the hierarchical tree. A holarchy thus represents a nested organizational structure, where the partial autonomy of the members typically allows them to handle transferred tasks in a flexible way and react to changing conditions without further coordination. This fact makes it distinguishable from a hierarchy in which agents are typically completely dependant on their predecessors. As further difference, a holarchy can also specify cross connections between members, thus dissolving the strict hierarchical structure.

Coalitions A coalition-based organization arranges the agents of a multiagent system into one or more coalitions which results in a global coalition structure (cf. Section 1.2). Coalitions are generally goal-directed in the sense that they serve the fulfillment of a specific objective and dissolve after this objective has either been met or lost its relevance. While internally the members of a coalition coordinate their actions in order to achieve the pursued goal, from an external point of view they appear as a single, integrated entity which is represented by a designated agent. However, as the representative has no authority with regard to the control of other members, a coalition's structure is generally flat. With respect to their membership, agents can either be part of one or multiple coalitions at a time. In the latter case, the coalitions of a coalition structure thus potentially overlap.

Societies A society represents an open, long-lived organizational structure which provides its members a framework for interaction. To this end, it prescribes global rules like binding communication protocols, ontologies, or social conventions which facilitate the reasoning of the participants by restricting their scope of action. Thus, a society defines a regulated environment in which heterogenous agents can pursue their individual goals and arrange themselves into further suborganizations if required. A compliance with the global rules can be encouraged by means of appropriate mechanisms like reputation systems which penalize a deviating behavior or enforced through third-party institutions which validate the members' activities.

Federations Similar to a coalition-based organization, a federation consists of different agent groups which interact via designated representatives. However, in contrast to coalition members, group members do not interact with each other but only with

their representative which acts as intermediary coordinating local and global activities. More precisely, a representative is able to receive task requests from other intermediaries and allocate them to members of its group based on descriptions of their capabilities. Vice versa, internal task requirements are matched with the competencies of other groups and sent to their corresponding intermediaries. Furthermore, a representative can also keep track of local activities and block undesired external communication. As the agents of different groups do not interact directly, federations are particularly suitable for the integration of proprietary systems which functionality can then be wrapped through appropriate intermediaries.

As can be seen from the diversity of the above examples, the organizational structure of a multiagent system has to be chosen according to the requirements of the given application. For instance, hierarchies and holarchies are particularly suitable for use cases in which problems are decomposable into smaller subproblems and the resulting decomposition structure can be mapped to a tree. As justified in more detail in Section 2.4, for the multiagent system of this thesis we choose a coalition-based organization because it allows for a flexible, product-based pooling of energy units and thus for the achievement of our goals as defined in Section 1.3.

2.3 Design of Experiments

The following sections provide an introduction to Design of Experiments. As a comprehensive approach for experimentation, it comprises a broad range of different techniques which can be applied according to the given requirements[1]. While Section 2.3.1 provides an overview of the basic concepts, Section 2.3.2 describes the experimental designs which we apply for the evaluation of our new aggregation method in Chapter 5. Afterwards, Section 2.3.3 and 2.3.4 detail aspects of effect calculation and response surface modeling, respectively. The descriptions mainly draw on [NIS13, Kle13, AW07, MMAC09, Kle08].

2.3.1 Basic Concepts

Design of Experiments is an approach for conducting experiments in an efficient way. This is achieved through the application of techniques which maximize the gained information while minimizing the experimental efforts. Originating in the domain of agriculture in the 1920s, DOE was first mainly applied in process industries but also attracted more and more attention from other disciplines like engineering or psychology. Although originally developed for real (i.e. non-simulated) experiments, the concepts from DOE can generally be applied to any system with measurable inputs and outputs in order to examine cause and effect relationships. Still, special attention has to be paid when dealing with deterministic processes like computer simulations. As these are not exposed to stochastic influences and thus yield the same output if reexecuted under the

[1] Alternatively, DOE can be viewed as a domain providing concepts for efficient experimentation.

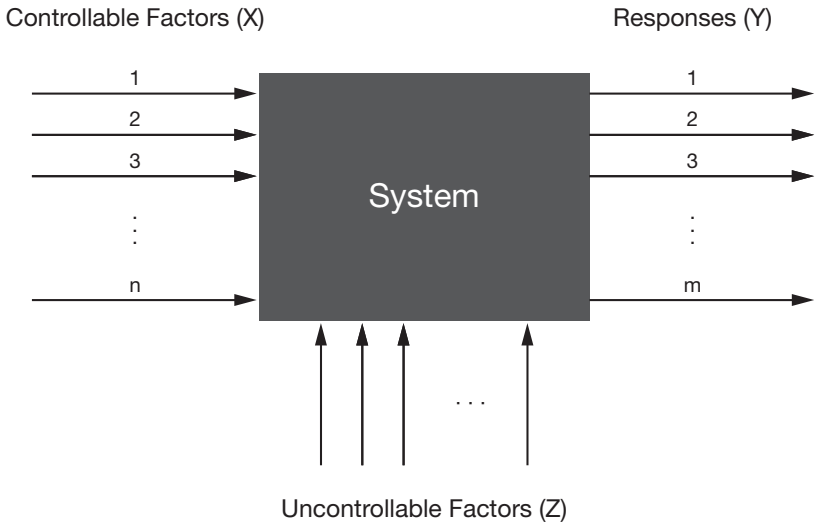


Figure 2.8: DOE black box model (adopted from [NIS13]).

same conditions, not all of the original DOE concepts are reasonably applicable. Because we apply the approach for the analysis of deterministic simulations, we discuss related issues where required.
Figure 2.8 shows the experimental model which is applied in the context of DOE. As shown, the system under investigation is considered as a black box which can be deliberately influenced by means of a number of controllable factors X (also referred to as input parameters). Moreover, it can be implicitly affected by a set of uncontrollable factors Z, like ambient temperature or humidity. However, note that in computer simulations uncontrollable factors do not exist unless they are explicitly modeled. The investigated characteristics of the system are finally given by a set of measurable responses Y [AW07, NIS13]. Given this model, the purpose of DOE is to asses causal relationships between a set of considered factors X and responses Y through systematic experimentation. The gained information is used in order to optimize the system with regard to a pursued goal, like the maximization of a specific response or the improvement of the system's robustness. Generally, the approach can be applied to a broad range of problems which can be classified into the following categories [NIS13]:

Comparative choice The first class covers problems in which a choice has to be made between two or more different alternatives (e.g. between two different parts in a production process). Thus, there is only one factor which is investigated with regard to its alternative effects on a considered response. The experiments can either be conducted under equal conditions for an initial comparison or under varying conditions in order to achieve more reliable results.

Factor Screening Typically not all of a set of controllable factors are of equal importance with regard to their effect on an examined response, in particular if a high number of parameters is considered. Thus, goal of experiments related to this class is to assess the impact of all parameters and screen out the most influential ones in order to focus on these in the course of the further optimization process.

Response surface modeling (RSM) A response surface reflects the output behavior of a system with regard to an examined characteristic given a set of controllable factors X. Accordingly, experiments related to this category allow to fit polynomial models which describe the local surface of a response and can thus be used for the achievement of a pursued optimization goal. As the complexity of these models increases with the number of involved factors, a screening is typically conducted beforehand if their number exceeds a manageable quantity. Response surface models for instance allow to hit a specific target value, maximize output, make a process more consistent by reducing variation, or make a system more robust against external influences.

Mixtures In mixture problems, the controllable factors are proportions of a blend which sum up to a specific value. General goal of respective experiments is to find a setting which is optimal with regard to a considered response.

Regression modeling In some cases response surface modeling is not sufficient as polynomial functions approximate a response surface only over a restricted region. If a more detailed description is required for a given response, other experimental techniques have to be applied in order to create a more sophisticated mathematical model.

As the above descriptions show, DOE is a versatile approach for experimentation. Besides its broad applicability, its success also stems from the advantages it provides over other experimental methods. Particularly compared to the widely used one-factor-at-a-time (OFAT) approach which only varies one parameter while keeping all others fixed when examining a response, DOE provides some key benefits with regard to experimental reliability and efficiency. A pivotal DOE concept which allows for these advantages is that of a factorial design. To illustrate, Figure 2.9 shows for both approaches the design space of an experiment in which two factors are varied between a low ($-$) and a high level ($+$). As described in more detail below, in the context of DOE the effect of a factor on a response is generally determined by contrasting level-related response averages, i.e. by calculating the difference between the response average of the low levels and the response average of the high levels. Given this technique, DOE provides the following advantages over the OFAT approach [AW07, Kle13]:

Efficiency Through the use of factorial designs, DOE requires less simulation runs than OFAT. For instance, in the example in Figure 2.9, DOE requires four runs in order to asses the effect of factor A and B. In contrast, OFAT requires six runs in order to determine effects of similar expressiveness. Generally, the efficiency of DOE increases the more, the more factors are involved. For example, in case of three parameters DOE requires eight runs, whereas OFAT requires sixteen runs in order to generate similar results.

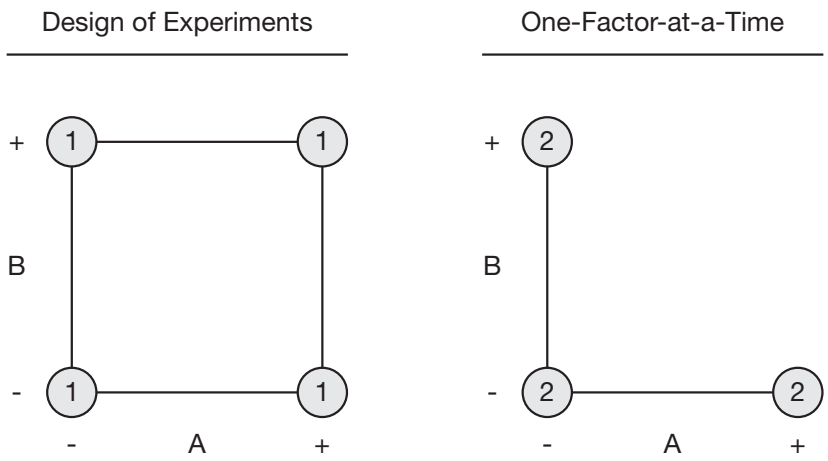


Figure 2.9: Design space of a DOE and OFAT experiment (adapted from [Kle13]).

Inductive Basis The design space of a factorial design allows to infer results from a wider inductive basis. For instance, when conducting an OFAT experiment as reflected by the design space on the right hand side of Figure 2.9, no information can be gained about the output when both factors are set to the high level. Moreover, the runs in which both factors are set to the low level is given to much weight as they are used twice for the calculation of the response averages.

Interactions As factor level combinations are integral part of a factorial design, DOE allows to identify interactions between parameters and thus to reveal new and often crucial information about a considered system. This kind of information can not be retrieved via OFAT as the approach completely omits combinations of factor levels.

Besides factorial designs, there are also other types of experimental designs which are applied in the context of DOE. An overview is given in Section 2.3.2.
DOE is typically applied using a structured process which comprises the following steps (adapted from [Kle13, NIS13]):

Step 1 – Definition of Objectives As first activity, the goal of the conducted experiment is specified which can typically be assigned to one of the above discussed categories. If several objectives are pursued, these have to be prioritized as the choice of the examined factors and the experimental design depends on the goal.

Step 2 – Parameters Selection The second step comprises the selection of all factors and responses which are considered relevant for the achievement of the previously specified goal. Moreover, the levels of each factor are determined which represent the settings to which the parameter is set in the course of experimentation. The actual number of levels which has to be defined depends on the pursued experimental goal. Generally, it is important to include all factors and responses which are relevant for

the considered context and choose factor settings which cover a wide operational range but do not lead to invalid configurations.

Step 3 – Design Specification Having identified the factors and responses which are to be examined, the third step consists of the specification of the actual experimental design. As discussed later in Section 2.3.2, the selection of the design type depends on the experimental goal, the number of examined factors, as well as the desired precision of the results. Moreover, if the system under investigation is exposed to stochastic effects, the design has to be accordingly specified in order to reduce random influences.

Step 4 – Design Execution In the fourth step, the experimental design is executed by conducting the specified runs and capturing the output. As the latter forms the basis for all conclusions drawn from the experiment, it is important to keep track of all unforeseen events and known but uncontrollable influences which have impacts on the results. If documented, these can be potentially identified in the course of the analysis of the outcomes.

Step 5 – Results Analysis The results gained from the execution of an experimental design can be analyzed by means of a broad range of graphical and numerical techniques depending on the pursued goal which was specified at the beginning of the process. While some objectives only require the construction of data plots, other aims involve the creation of empirical models which describe the behavior of the system under investigation. However, in order to prevent misleading conclusions, these models have to be validated with regard to their capability to describe the observed data correctly.

Step 6 – Documentation and Application of Results In the last step of the process, the results from the previous steps are documented and used in order to achieve the pursued goal. If an optimal setting was identified and the examined system is exposed to stochastic effects, the result is typically confirmed by executing the process several times based on the configuration.

Having introduced the basic ideas of DOE, in the following we provide a more detailed description of those concepts which are applied in the context of evaluation in Section 5.

2.3.2 Experimental Designs

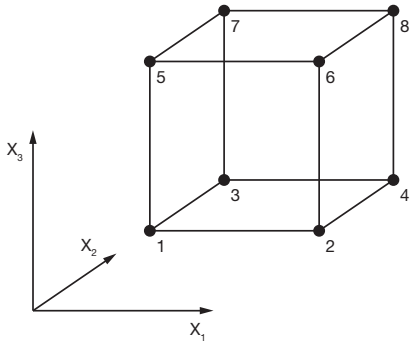
A key activity in the context of Design of Experiments is the selection of the actual experimental design. The choice particularly depends on the type of the given problem, the number of factors, as well as the considered system itself because it determines the complexity of an experimental run. Table 2.1 gives an overview of common experimental designs classified by the above discussed categories (for mixture problems and regression modeling mixture and regression designs are required, respectively). Generally, the

comparative choice	factor screening	RSM
completely randomized randomized block	full factorial fractional factorial Plackett-Burman	central composite Box-Behnken

Table 2.1: Overview of experimental designs (adapted from [NIS13]).

listed types allow to examine different numbers of factors for a given number of experimental runs. For instance, fractional factorial designs permit to handle a higher number of factors than full factorial designs[2]. As shown, for comparative choice problems either completely randomized or randomized block designs are used in order to investigate the effect of an examined factor. While the former are applied when there exist no nuisance factors – i.e. parameters which influence the output besides the examined factor –, the latter try to eliminate this kind of impacts by varying the considered factor within blocks in which all controllable nuisance factors are constant. The listed designs for factor screenings mainly differ in the number of experimental runs which have to be conducted for a given number of factors. In this regard, full factorial designs require on the one hand the highest number of runs, but provide on the other hand the most precise results by taking all possible factor level combinations into account. Thus, these are typically applied when a low number of factors is given. In contrast, fractional factorial designs allow to reduce the number of experimental runs by executing only parts of the corresponding full factorial designs according to the intended goal. Plackett-Burman designs are even more economical with regard to the required runs but are restricted to the assessment of main effects (i.e. effects which are caused by single factors) where it is assumed that all interaction effects are negligible. The final category in Table 2.1 comprises designs which are suitable for the purpose of response surface modeling. If a linear model is sufficient for the achievement of the experimental goal, also full factorial and fractional factorial designs can be applied in this context. However, often a non-linear function is required in order to approximate the surface of a response appropriately. In this case, designs have to define more than two levels for each factor in order to allow for a fitting of higher-order polynomials. Both central composite designs (CCDs) and Box-Behnken designs account for this requirement. The two types mainly differ in the way in which they define the different factor level combinations which thus results in different design spaces [NIS13, Kle13].
In the context of evaluation in Chapter 5, we investigate the algorithmic characteristics of our proposed aggregation method by conducting a factor screening and using the

[2] Because it generally holds that a higher number of input parameters requires a higher number of experimental runs, the actual number of factors which can be examined by means of a specific design strongly depends on the system under investigation. For instance, in the context of an industrial process five factors may be considered as a high number of input parameters, whereas in the context of a computer simulation the same number may be handled with ease as more experimental runs can be executed by the system with less effort.



std	X_1	X_2	X_3
1	−	−	−
2	+	−	−
3	−	+	−
4	+	+	−
5	−	−	+
6	+	−	+
7	−	+	+
8	+	+	+

Figure 2.10: Design space and matrix of a full factorial design (adapted from [Kle13]).

most important parameters in order to model the response surface of different evaluation criteria (i.e. responses), like the global performance or the runtime. To this end, we use full factorial and central composite designs which are described in more detail in the following sections.

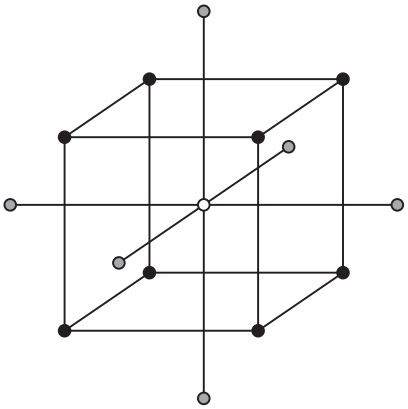
2.3.2.1 Full Factorial Designs

As already mentioned above, full factorial designs allow to screen a system's input parameters based on their effects and to fit first-order models which approximate the surface of a considered response. In the following, we provide an overview of the main characteristics of this design type and describe the calculation of effects and the modeling of response surfaces in Section 2.3.3 and 2.3.4, respectively [AW07, NIS13].
In general, full factorial designs are complete in the sense that they specify all possible factor level combinations. A design which includes k factors being set to n levels thus comprises n^k factor level combinations which results in a high experimental effort in terms of the required number of experimental runs. Hence, in order to reduce complexity, typically two factors levels are defined, i.e. a low level (also denoted as -1 or $-$) and a high level ($+1$ or $+$). The selection of the levels is generally a crucial task as an effect of a factor on a response may only be significant if these are reasonably set. As general rule, in two-level designs the low and the high level should thus be set to extreme values in order to induce an effect as high as possible.
To illustrate, Figure 2.10 shows the design space and design matrix of a two-level full factorial design with three factors. As depicted, the former is given by a cube, where the arrows indicate the direction in which each parameter is increased from the low to the high level. Each corner of the cube defines a factor level combination which can also be found in the corresponding row of the design matrix. The latter lists all possible combinations in standard order, i.e. a systematic sequence which has to be randomized in an experiment if the system under investigation is exposed to non-controllable, time-related effects in order to avoid misleading results. For instance, if the depicted design is executed in the specified order, in the first runs X_3 is solely set to the low level and

then solely set to the high level. If one of the levels has an uncontrollable impact on the process, the experimental results are consequently biased by this influence. However, if a process is completely deterministic (like a computer simulation), experiments can be run in standard order or in fact any other sequence without risking negative side effects.

2.3.2.2 Central Composite Designs

As described in the previous section, in full factorial designs factors are typically set to two levels only in order to reduce complexity. However, if the experimental goal is to approximate the surface of a response, this approach solely allows for the fitting of linear, first-order polynomial functions (for a more detailed description of RSM see Section 2.3.4). If the actual surface of the response has a strong curvature, though, these are inappropriate and higher-order functions are required in order achieve a sufficient goodness of fit. In this case, at least three levels have to be defined for each considered factor. While three-level full factorial designs generally satisfy this requirement, these are typically not efficiently applicable as the number of experimental runs increases exponentially in the number of factors (i.e. k factors entail 3^k runs). In contrast, three-level fractional factorial designs allow to reduce the number of runs, but have in this case several drawbacks which result from the increased number of factor levels (for a more detailed description see [NIS13]).
An appropriate design type for response surface modeling is given by a central composite design which allows to determine a higher number of levels while keeping the number of experimental runs in manageable orders. This is achieved by systematically adding additional points to the design space of a full factorial or fractional factorial design. To illustrate, Figure 2.11 shows the design space and matrix of a central composite design which extends the full factorial design of Figure 2.10. As depicted, the latter is augmented by center points (white) and star points (gray) which corresponds to the definition of five levels for each of the three included factors. The design matrix of the central composite design is made up of three parts which specify the previously discussed full factorial design (line 1-8), the star points (line 9-14), and the center points (line 15-18). The values (i.e. locations) of the star points and the number of center points are generally determined according to the experimental goals and the specific properties which the design is intended to possess (a description for the calculation of α can be found in [NIS13]). Moreover, if the examined factors can only be set to levels which are incompatible with the standard levels, the design can be adjusted accordingly [Kle13].
The above described type of central composite design is referred to as ‘circumscribed’ as the star points define new extreme values for the factors. However, if the low and the high levels which are defined in the full factorial design are true limits of the system which can not be exceeded, a central composite design can also be scaled down by setting the star points to the limits and adapting the levels of the factorial design accordingly (type ‘inscribed’). As final variant, the star points can also be located in the middle of the faces of the full factorial design space in which case $\alpha = \pm 1$ (type ‘face centered’) [NIS13, Kle13, AW07].



std	X_1	X_2	X_3
1	−	−	−
2	+	−	−
3	−	+	−
4	+	+	−
5	−	−	+
6	+	−	+
7	−	+	+
8	+	+	+
9	$-\alpha$	0	0
10	α	0	0
11	0	$-\alpha$	0
12	0	α	0
13	0	0	$-\alpha$
14	0	0	α
15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0

Figure 2.11: Design space and matrix of a central composite design (cf. [Kle13]).

2.3.3 Effect Calculation

Executing one of the screening designs in Table 2.1 yields response-related results for each factor level combination specified in the design matrix. These can be used in order to asses the effects of the considered factors and their interactions on the responses. A factor interaction exists if a factor's effect depends on the level of one or more other factors, and vice versa. I.e., interactions can occur between arbitrary combinations of factors, where for each response of a two-level design with k factors there are generally $2^k - 1$ assessable main and interaction effects.
With regard our exemplary full factorial design shown in Figure 2.10, this means that we can determine the main effects of the factors X_1, X_2, and X_3 as well as the interaction effects of the interactions X_1X_2, X_1X_3, X_2X_3, and $X_1X_2X_3$. As already outlined at the beginning of this introduction, in case of two-level designs the effects of single factors can be calculated by contrasting the response averages which relate to the low and the high levels. More precisely, the effect of a factor X is calculated as

$$eff_X = \frac{\sum Y_+}{n_+} - \frac{\sum Y_-}{n_-}, \tag{2.1}$$

	factor			interaction			
std	X_1	X_2	X_3	X_1X_2	X_1X_3	X_2X_3	$X_1X_2X_3$
1	−	−	−	+	+	+	−
2	+	−	−	−	−	+	+
3	−	+	−	−	+	−	+
4	+	+	−	+	−	−	−
5	−	−	+	+	−	−	+
6	+	−	+	−	+	−	−
7	−	+	+	−	−	+	−
8	+	+	+	+	+	+	+

Table 2.2: Analysis matrix of a 2^3 two-level full factorial design (adapted from [AW07]).

where Y_+ and Y_- are the individual response values and n_+ and n_- are the level-related numbers of runs. With regard to our example, the effect of factor X_1 can for instance be calculated as

$$eff_{X_1} = \frac{Y_2 + Y_4 + Y_6 + Y_8}{4} - \frac{Y_1 + Y_3 + Y_5 + Y_7}{4}. \tag{2.2}$$

Interaction effects are calculated in the same fashion, where the level of an interaction is determined by calculating the product of the levels of the involved factors. With regard to our exemplary design, Table 2.2 gives an overview of all factor and interaction levels. For instance, the levels of interaction X_1X_2 are calculated by multiplying the column of factor X_1 and X_2 [AW07, NIS13, Kle13].
If the system under investigation is exposed to stochastic influences, from a statistical point of view the response values which are used for the calculation of an effect can be considered as a random sample which is drawn from an infinite population (the infinite set of all possible, randomly influenced outputs). In this case, it is necessary to assess the significance of an effect in order to exclude the possibility that it was caused by mere coincidence. This can be achieved through confidence intervals which provide an interval estimation of an unknown parameter with regard to a specific confidence level (e.g. 95 % or 99 %). More precisely, a confidence interval allows the conclusion that if an experiment is conducted repeatedly and an interval is constructed for each run, the percentage of the intervals that includes the true value of the unknown parameter is equal to the defined confidence level. For instance, suppose we examine the mean of a population's parameter and calculate confidence intervals for different random samples (in which case we can obtain different intervals for different samples as these are chosen on a random basis). Then a confidence level of 95 % allows us to conclude that 95 % of the constructed intervals include the real value of the mean.

With regard to a considered effect eff, a confidence interval CI is generally calculated as

$$CI = eff \pm t \cdot SE, \tag{2.3}$$

where t is a t-value and SE is the standard error of the effect. While a more detailed description of t and SE can for instance be found in [AW07], for our discussion it is sufficient to note that a t-value depends on the desired confidence level as well as the degrees of freedom available for the estimation of SE which in turn are determined by the number of conducted experimental runs. It generally holds that the higher the confidence level and the lower the number of runs, the higher the t-value. In contrast, the standard error of the effect depends on the estimated variance of the results as well as the number of experimental runs, where the value becomes smaller if either the variance becomes smaller or the number of runs becomes higher. It follows that the width of a confidence interval can be influenced by the chosen confidence level as well as the number of experimental runs. More precisely, a smaller interval can be obtained by decreasing the confidence level or increasing the number of runs, and vice versa.
If a confidence interval is now constructed for a calculated (i.e. observed) effect, the basic idea is that if the interval includes the value 0, the effect can not be considered significant because the real effect might be 0 and the calculated effect may just has been caused by mere coincidence. However, if the interval does not include 0, we can be confident that the effect was in fact caused by the related factor or interaction. In practice, 95 %, 99 %, and 99.9 % confidence intervals are used in order to identify indifferent, significant, and highly significant effects, respectively [AW07, Kle13].
However, in this context attention has to be paid if a full factorial design is applied and the examined system is deterministic. As described in Section 2.3.2.1, the former comprises all possible factor level combinations, while a deterministic system always yields the same output for a given design if it is repeatedly executed. From a statistical point of view, this means on the one hand that the samples are drawn from a finite population (as the process is deterministic) and on the other hand that these cover the whole population (as all possible factor level combinations are taken into account). It follows that in this case the construction of confidence intervals is not reasonable as in fact there is no randomness, neither with regard to the considered system nor with regard to the drawing of the samples. Thus, if a calculated effect is unequal 0, it can be concluded that there is indeed an effect induced by the related factor or interaction. Consequently, in order to identify which effects are important, in this case an alternative decision rule has to be defined.
The effects which result from the execution of an experimental design can be visualized by means of a Pareto chart in which these are depicted as a (typically ordered) series of bars and possible confidence intervals are shown as horizontal, symmetric lines. As an example, the chart in Figure 2.12 shows the effects of four factors and their two-way interactions. An effect is significant with regard to a specific confidence level if its bar reaches the corresponding line.

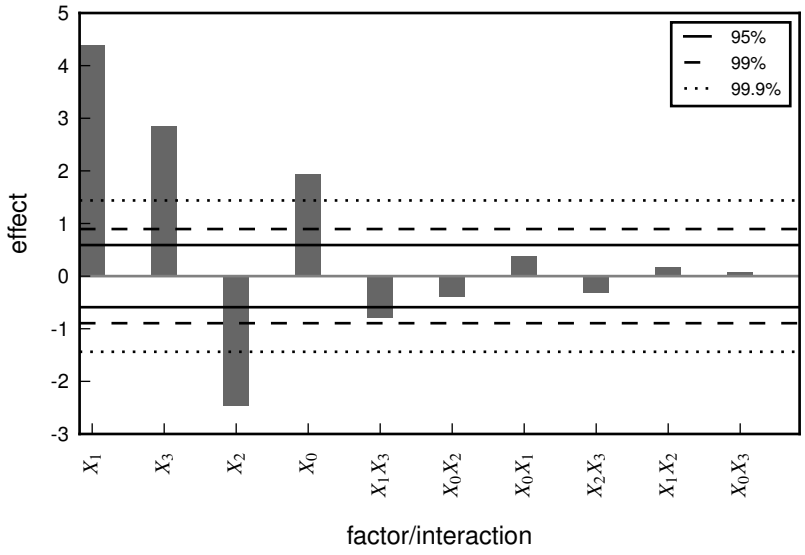


Figure 2.12: Pareto chart plotting effects and confidence intervals.

2.3.4 Response Surface Modeling

As already mentioned above, the results obtained from the execution of an experimental design can also be used in order to create a model which approximates the surface of a considered response Y. Generally, the model's degree of approximation to the true surface depends on several factors, like the amount of collected data or the number of levels defined for each factor. In this regard, it generally holds that the more information is obtained through experimentation (i.e. the more experimental runs are conducted and the more factor levels are specified), the better the approximation.
A common way to model the surface of a response is through the fitting of a polynomial function which describes the surface over a restricted region [MMAC09, NIS13, Kle08]. The simplest form of this kind of function is given by a first-order polynomial which specifies a linear relationship between a response Y and k factors X_j, i.e.

$$\begin{aligned} Y &= \beta_0 + \sum_{j=1}^{k} \beta_j X_j + \varepsilon \\ &= \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + \varepsilon, \end{aligned} \tag{2.4}$$

where β_0 is the intercept, β_j is the coefficient of variable X_j, $1 \leq j \leq k$, and ε is an error term which represents sources of variability not described by the prior linear term. It is typically assumed that ε is normally distributed with mean 0 and variance σ^2. If all other variables are kept fixed, each coefficient β_j measures the change in Y per unit change in X_j.

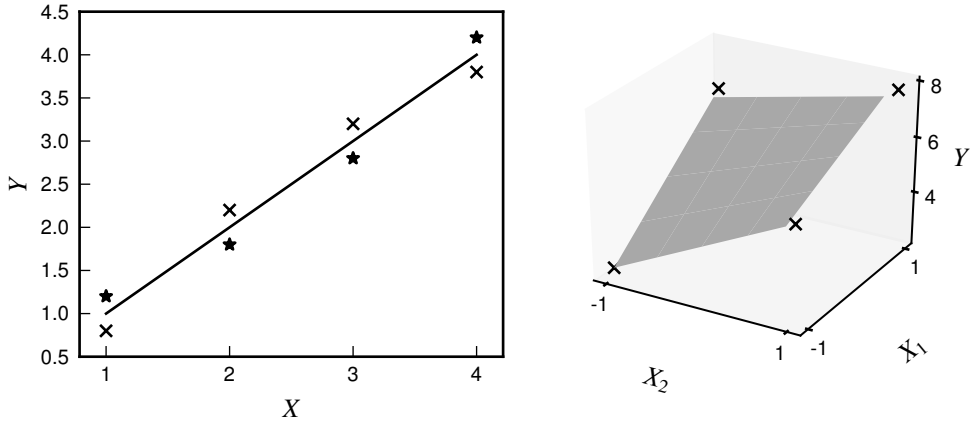


Figure 2.13: First-order model with one factor (left) and two factors (right).

Models of the above form are generally referred to as linear regression models with regression coefficients β_j and regressor variables X_j. If these are chosen in order to approximate the surface of a response (meaning that the output behavior of the system is assumed to be accordingly describable), regression analysis can be applied in order to estimate the unknown coefficients β based on the data obtained from the execution of an experimental design. For instance, assume that we investigate the response of a stochastic system with a single input parameter which can be set to four different levels. Further suppose that we execute an experimental design (in this case a completely randomized design) which specifies two runs for each factor level (i.e. eight runs in total) and yields the results shown on the left hand side of Figure 2.13. I.e., in terms of Equation 2.4 we obtain a result Y_i for each experimental run $i = 1, \ldots, 8$ which enables us to define a corresponding number of polynomial functions. In general, for a design with n runs we can specify a system of equations as follows:

$$\begin{aligned} Y_i &= \beta_0 + \sum_{j=1}^{k} \beta_j X_{ij} + \varepsilon_i \\ &= \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_k X_{ik} + \varepsilon_i, \quad i = 1, \ldots, n. \end{aligned} \tag{2.5}$$

Given the above system, the method of least squares can be applied in order to estimate the coefficients of the final regression model [Sta13, Kle13, MMAC09]. This is achieved by determining the β's in a way which minimizes the sum of the squared errors ε_i^2. With regard to our example shown on the left of Figure 2.13, this approach fits a straight line

to the observed data points such that the squared deviations of the points from the line are minimized. Formally, the minimized least squares function is defined as

$$\begin{aligned} L &= \sum_{i=1}^{n} \varepsilon_i^2 \\ &= \sum_{i=1}^{n} \left(Y_i - \beta_0 - \sum_{j=1}^{k} \beta_j X_{ij} \right)^2 . \end{aligned} \tag{2.6}$$

I.e., the method of least squares minimizes L with respect to the coefficients β_i. The fitted model resulting from this process is defined as

$$\hat{Y} = b_0 + \sum_{j=1}^{k} b_j X_j, \tag{2.7}$$

where $b_0, \ldots, b_k$ are referred to as least squares estimators. The differences between the observed response values Y_i and the predicted values of the fitted model $\hat{Y}_i$ are termed residuals e_i. I.e., for an experimental design with n runs, these are simply calculated as

$$e_i = Y_i - \hat{Y}_i, \quad i = 1, \ldots, n. \tag{2.8}$$

With regard to the line plot in Figure 2.13, the residuals are thus the distances between the observed points and the regression line. As estimates of the experimental error ε, they are also assumed to be randomly distributed with mean 0 and variance σ^2. Residuals are typically used in order to evaluate a model's goodness of fit as described in more detail at the end of this section.

Generally, linear regression models of the above form describe a hyperplane in a k-dimensional space which is defined by the variables X_j. As shown in Figure 2.10, in case of $k = 1$ and $k = 2$ factors the described response surface can be visualized as a line and plane, respectively. In the latter case, the plane is fit to the observed data points which lie in the three-dimensional space. Note, however, that functions as described by Equation 2.4 do not allow to model any curvature as they do not take any factor interactions into account. The latter can be considered by adding respective terms to the equation. For instance, a first-order polynomial with two-way interactions is defined as follows:

$$Y = \beta_0 + \sum_{j=1}^{k} \beta_j X_j + \sum_{j=1}^{k-1} \sum_{j'=j+1}^{k} \beta_{jj'} X_j X_{j'} + \varepsilon. \tag{2.9}$$

I.e., for any two factors X_j, $X_{j'}$, the function includes a term $X_j X_{j'}$ describing their interaction. The coefficients can again be estimated using the method of least squares.

However, although factor interactions introduce curvature into the model, the shape of the true response surface is often too rough and can not be appropriately approximated by first-order polynomials. For instance, a two-way interaction may bend the shape of a

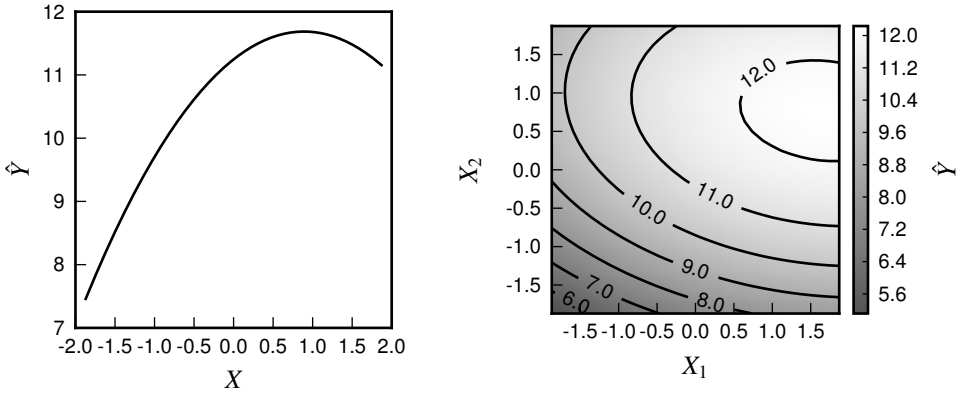


Figure 2.14: Line plot and contour plot describing non-linear relationships.

plane, but does not allow to account for stronger curvatures like a peak in the middle of the surface. This requires the fitting of higher-order models which can be created based on dedicated experimental designs as discussed in Section 2.3.2. For instance, second-order polynomial functions add purely quadratic terms to Equation 2.4 which allows for a better approximation. Formally, they are defined as:

$$Y = \beta_0 + \sum_{j=1}^{k} \beta_j X_j + \sum_{j=1}^{k} \sum_{j' \geq j}^{k} \beta_{jj'} X_j X_{j'} + \varepsilon. \tag{2.10}$$

Although the above equation includes a quadratic term X_j^2 for each factor X_j, linear regression can be applied in order to identify the least squares estimators $b_1, \ldots, b_k$ as the function is linear in its coefficients β. Figure 2.14 shows two exemplary second-order models including one and two factors. While the line plot on the left describes a non-linear relationship between a factor X and a predicted response $\hat{Y}$, the contour plot on the right depicts the impact of two parameters X_1 and X_2 on another response $\hat{Y}$, where the non-linearity can be seen from the color gradient and the contour lines.
In contrast to the approach described in Section 2.3.3, the effect of a factor or an interaction can also be determined using the related least squares estimator b provided that specific requirements are fulfilled [Kle13]. In this case, it simply holds that

$$b = \frac{eff}{2}. \tag{2.11}$$

For instance, the effect of a factor X_j with coefficient $b_j = 5$ can then be calculated as $eff = 2 \cdot 5 = 10$.
After identification of the least squares estimators, it is necessary to validate a model in order to evaluate how well it fits the observed data (and thus how properly it approximates the true response surface). In this context, typically both numerical and graphical

techniques are applied which use the residuals of a model in order to verify its goodness of fit. A common numerical method is the coefficient of determination R^2 (also referred to as R-Squared) which quantifies the fraction of the variability in the response being explained by the model [Sta13]. More precisely, a high R-Squared indicates that the residual variability relative to the variability in the response is small and thus that the model fits the data well because it provides predictions not causing extra variability in the residuals. It generally holds that $0 \leq R^2 \leq 1$, where an R-Squared of 0.8 means for instance that 80 % of the variability in the response are explained by the model and 20 % relate to the variability in the residuals.
However, a high R-Squared is no guarantee for a good fit as it does not account for all relevant aspects. Thus, a model is typically further validated by means of a graphical residual analysis as exemplary shown in Figure 2.15. The different data plots allow to examine the following aspects:

Normal distribution As mentioned above, the residuals are typically assumed to be normally distributed. In order to validate this property, they are visualized in the form of a probability plot which compares two probability distributions by plotting the quantiles of the residuals against the theoretical quantiles of the normal distribution. The x-axis of the plot is adjusted in such a way that the residuals lie on a straight line if following a normal distribution. Deviations from the line thus indicate deviations from a normal distribution. An exemplary probability plot is depicted in the upper left corner of Figure 2.15 which shows that the residuals are approximately normally distributed.

Independence from factor levels If the fitted model describes the observed data appropriately, the residuals do not depend on specific parameter settings. This is verified by plotting them against the factor levels as shown in the upper right corner of Figure 2.15. The residuals are independent from the levels if the points form a horizontal band.

Constant variance In order to verify the assumption that the residuals have constant variance and are independent from the predicted values of the fitted model, they are plotted against the latter. If the assumption holds, the points are randomly distributed about 0 as shown in the middle plot of Figure 2.15. However, often the residuals increase with the size of the predicted values in which case the original data can be transformed (for instance through a logarithm or a square root transformation) in order to achieve a better result [AW07].

Independence from time Temporal trends in the experimental results can be revealed by plotting the residuals against the runs of the executed design. If no trend exists, the points are randomly distributed about 0 as shown in the lower plot of Figure 2.15. Generally, a temporal trend can only be assessed via a residual analysis if the experimental run order was randomized (i.e. the design was not executed in standard order) as otherwise the results are biased by the very same. However, since deterministic

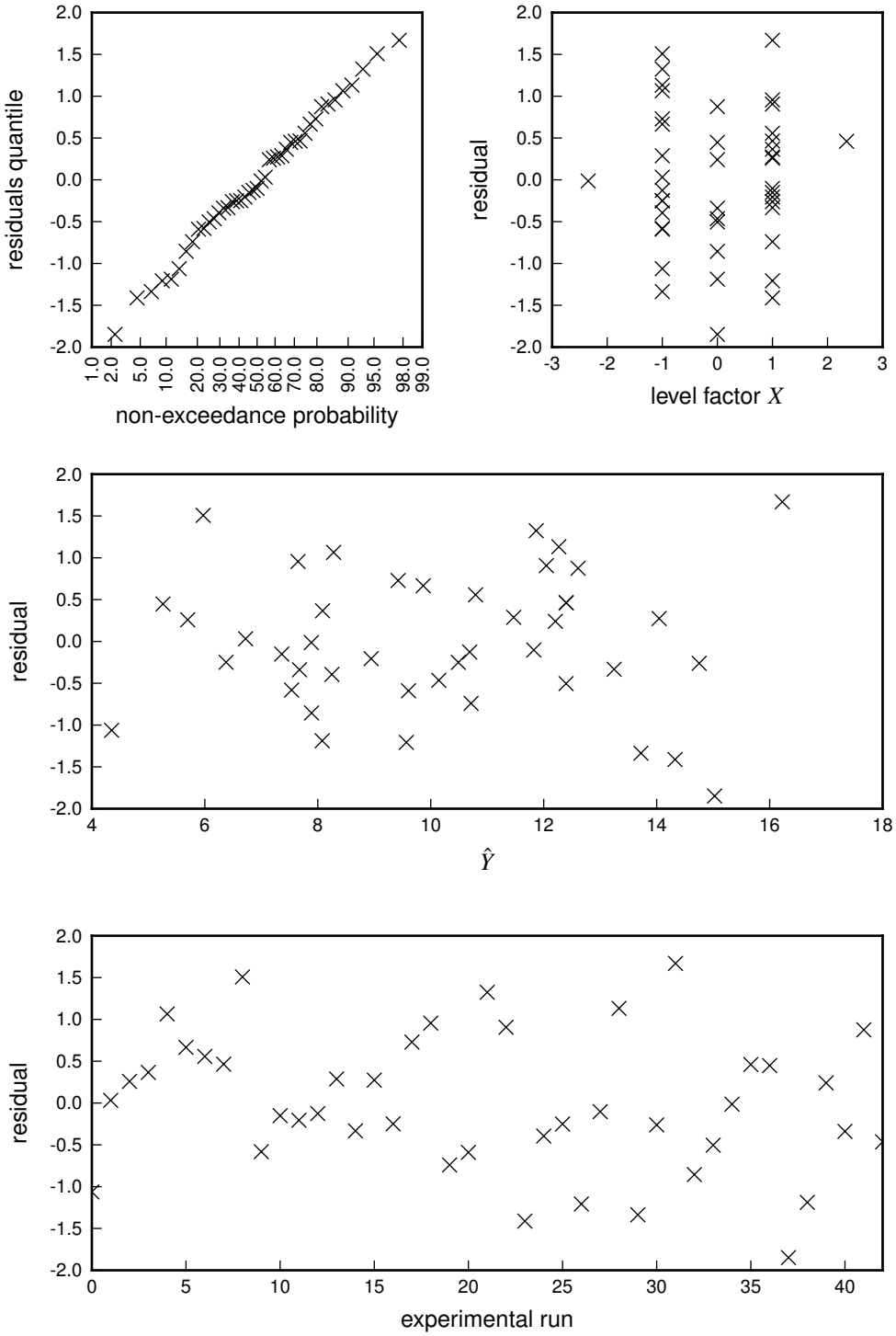


Figure 2.15: Exemplary residual analysis.

systems always yield the same results independent from the order of the runs, an identification is only reasonable in case of stochastic systems.

Any irregularity identified via the above graphical analyses indicates that the residuals contain structure which the fitted model does not account for. Adding appropriate terms to the model improves the fit and thus the approximation to the response surface [Kle13, NIS13].

2.4 Summary and Discussion

In the following we summarize the descriptions of the previous sections and discuss them in view of our objectives as defined in Section 1.3.
Section 2.1 first provided an overview of the deregulated electricity industry and outlined the processes which connect the different actors on the economic and physical level of the system. Afterwards, we discussed the structure of electricity markets as well as the different market types which can be applied in order to arrange trades between buyers and sellers. In this regard, we expect our new aggregation method to be universally applicable in the sense that it is independent from a specific market type and allows participants to consider products of different submarkets when optimizing their individual profit.
Section 2.2 then provided an introduction to the domain of distributed artificial intelligence and first discussed the concept of an intelligent agent along with potential characteristics of its environment. With regard to our goal to provide an agent-based approach for the distributed pooling of energy units in the context of electricity markets, from the viewpoint of a single agent the considered setting represents the most complex type of environment because it is

- *partially observable* as the agent has no complete information about the state of the other agents or their units,
- *stochastic* as the next state of the environment is also determined by the other participants in the market,
- *sequential* because the agent's behavior in the course of a pooling process has an impact on its trustworthiness and thus its success in future pooling processes,
- *dynamic* because pools are formed in a parallel fashion and changes occur while the agent is reasoning,
- *continuous* because characteristics of its controlled unit are associated with continuous scales,
- *unknown* because the agent has no complete knowledge about the outcomes of its formation activities, and
- a *multiagent* environment because the agent interacts with with other participants in order to form pools.

Having discussed the concept of an intelligent agent, we covered multiagent systems and detailed how agents communicate and coordinate in order to achieve a coherent overall behavior. Besides pointing out the different classes of cooperative and competitive coordination techniques, we also provided an overview of organizational structures which define the roles, authorities, and relationships in a MAS. In this context, we mentioned that we choose a coalition-based organization for our agent-based aggregation method as it allows for the fulfillment of the requirements and objectives as defined in Section 1.1 and 1.3, respectively. More precisely, as this organizational structure is characterized by a goal-directed formation of the coalitions, it enables a temporally flexible pooling of energy units for the joint fulfillment of single power products. Moreover, the coalitions' encapsulated nature and flat organizational structure allow for the design of a distributed interaction protocol in which the representatives act on behalf of their group.
Finally, Section 2.3 provided an introduction to Design of Experiments and its underlying concepts. With regard to the evaluation of our proposed approach in Chapter 5, we detailed full factorial and central composite designs which are used in order to identify the most important input parameters and model the surfaces of the considered evaluation criteria. In this regard, the chapter also covered the basics of effect assessment and discussed polynomial functions as common means for RSM.

3 Problem Formalization

Having covered the fundamentals which underly this thesis, in the following we continue our examinations by analyzing the considered problem in detail and creating a model referred to as DYCE-FM (DYCE Formal Model). By providing precise definitions of all relevant concepts and setting these into relation, the latter provides a sound basis for the upcoming design and evaluation of our new aggregation method. As we intend the method to be independent from both the type of a market and the type of an energy unit, all concepts are specified in such a way that they are applicable to the general use case of coalition formation in electricity markets. Moreover, the given definitions are abstract in the sense that they allow for different designs. For instance, we define an error with which a coalition produces or consumes electrical energy but do not specify how this error is actually modeled. Specifications which concern the design of our approach are given in the following chapter.
Figure 3.1 gives an overview of the general structure of DYCE-FM and its covered domains. The included definitions finally end in the specification of an *electricity market*, where the place of delivery is given by a *power grid* which connects a set of energy units being controlled by autonomous intelligent agents. The latter are able to cooperate and form *coalitions* for *value maximization*. General goal of a coalition is the trade of a power *product* which is physically fulfilled in a temporal *product horizon*. Through its joining, an agent makes a *contribution* to a coalition's *cumulative contribution* which fulfills a pursued target product if the specified conditions are met. Each contribution is determined by the *operation schedule* of an agent's unit, where the general time frame for scheduling is given by a temporal *planning horizon*.
Having outlined its general structure, in the following we specify DYCE-FM in detail by covering its four different domains in Section 3.1-3.4. In order to provide an orientation, Figure 3.2 gives an overview of the forty defined concepts and their basic relationships. Moreover, Table 3.3 lists all specified symbols with references to the respective definitions. Because coalition formation is also a well known problem in the fields of distributed artificial intelligence and game theory, we adopt definitions from these domains where appropriate.

3.1 Domain 1 – Power Grid, Coalition, Value Maximization

The first domain of DYCE-FM covers concepts which are related to the agents as well as their goal to cumulate the potentials of their controlled units and maximize value through coalition formation. Thus, the definitions also address aspects related to their reasoning and social behavior. As starting point, we begin with the specification of the electrotechnical infrastructure to which the units are connected.

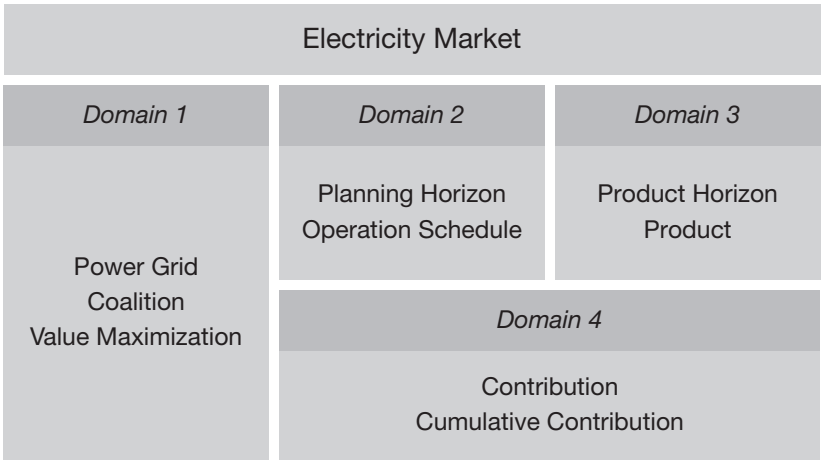


Figure 3.1: Structure and covered domains of DYCE-FM.

Definition 3.1: Power Grid

Let $G = (V_G, E_G)$ be a *power grid* which is represented as a weighted graph, where $V_G = \{v_{G,1}, \ldots, v_{G,n}\}$[1] is a set of vertices representing the *grid nodes* and E_G is a set of edges $\{v_{G,i}, v_{G,j}\}$, with $v_{G,i}, v_{G,j} \in V_G \wedge i \neq j$, representing the *power lines*. Each edge $\{v_{G,i}, v_{G,j}\}$ is assigned a *distance weight* $dw(v_{G,i}, v_{G,j})$, where a higher value reflects a greater distance.

A grid node $v_{G,k}$ represents electrotechnical equipment of a power grid (e.g. a house connection box), where all nodes of a grid are connected through power lines $\{v_{G,i}, v_{G,j}\}$. As described in more detail in the context of Definition 3.9, a line's distance weight $dw(v_{G,i}, v_{G,j})$ quantifies the distance between its endpoints and is derived from a set of dedicated criteria. According to its network topology, a power grid can be assigned to a specific class, like a radial or a meshed network.
In order to allow for the automation of a power grid, it has to be equipped with an appropriate ICT infrastructure which enables a physical exchange of information.

Definition 3.2: ICT Network

Let $ICT = (V_{ICT}, E_{ICT})$ be an *ICT network* which is represented as a weighted graph, where $V_{ICT} = \{v_{ICT,1}, \ldots, v_{ICT,m}\}$ is a set of vertices representing the *communication nodes* and E_{ICT} is a set of edges $\{v_{ICT,i}, v_{ICT,j}\}$, with $v_{ICT,i}, v_{ICT,j} \in V_{ICT} \wedge i \neq j$, representing the *communication lines*. Each edge $\{v_{ICT,i}, v_{ICT,j}\}$ is assigned a *quality weight* $qw(v_{ICT,i}, v_{ICT,j})$, where a higher value reflects a better transmission quality.

[1] In the following it generally holds that $\mathbb{N} = \{1, 2, \ldots\}$, $\mathbb{N}_0 = \{0, 1, 2, \ldots\}$, and, if not stated otherwise, all indices $i \in \mathbb{N}$.

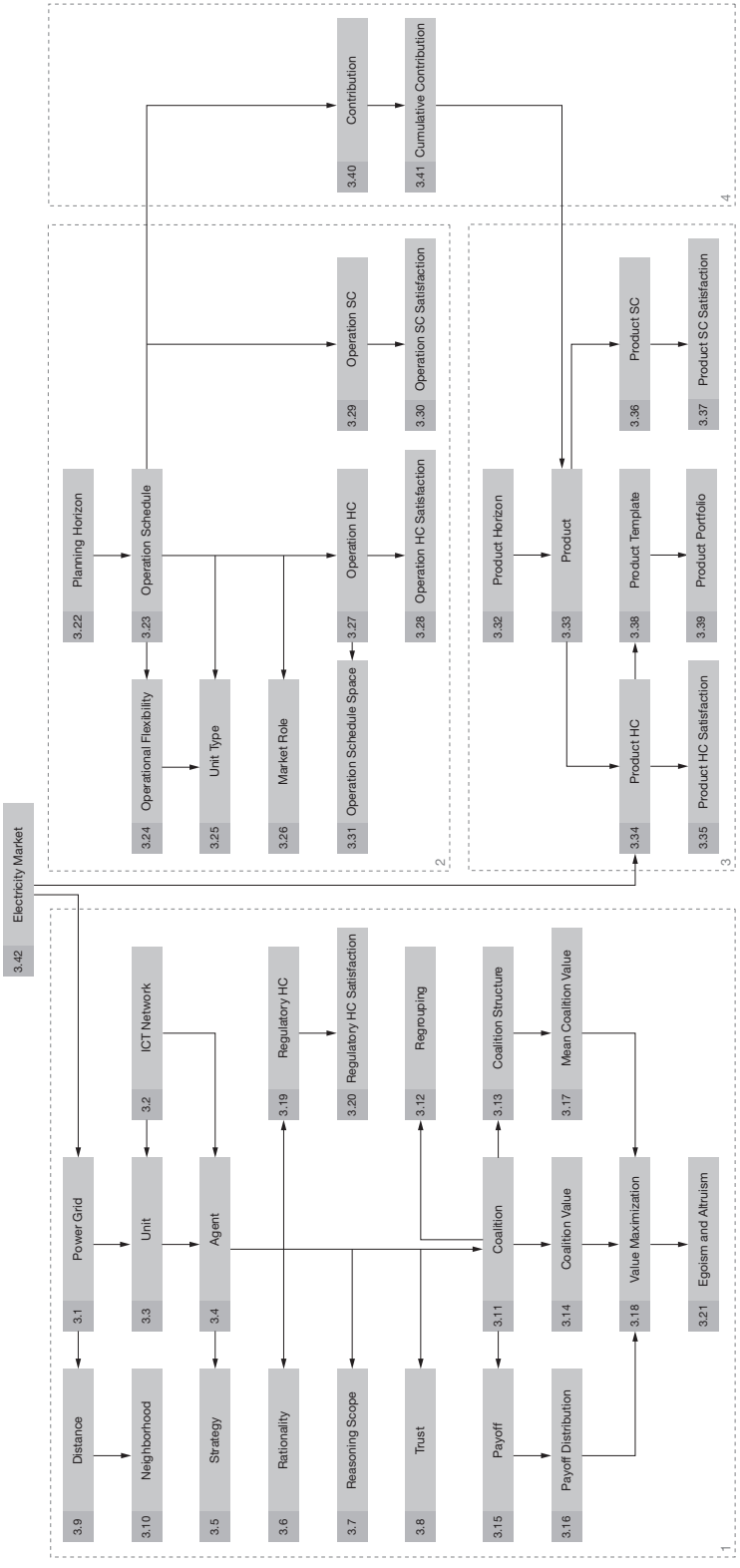


Figure 3.2: Basic relations between the definitions of domain 1-4.

A communication node of an ICT network is a device which is able to receive and send data and process it according to a given algorithm. The data are transferred based on a defined protocol, where the transmission quality can be compromised by different factors. Thus, the latter is reflected by quality weights which are attached to the communication lines, where a minimum weight reflects a total loss of data and a maximum weight indicates a perfect transfer.
The communication nodes of an ICT network can either be given by separate computer systems or embedded systems which are installed on energy units being connected to a power grid.

Definition 3.3: Unit

Given an ICT network ICT with $V_{ICT} = \{v_{ICT,1}, \ldots, v_{ICT,m}\}$, let $U_a = \{u_1, \ldots, u_n\}$ be a finite and nonempty set of *atomic units* u_i which are able to produce or consume electrical energy. Each atomic unit u_i is equipped with an embedded system, thus representing a node of ICT as determined by a *communication node assignment* $cn_U : U_a \to V_{ICT}$, with $cn_U(u_i) = v_{ICT,j}$, $1 \leq i \leq n$, $1 \leq j \leq m$. A *unit* $U \subseteq U_a$ is then defined as a finite and nonempty set of one or more atomic units, where the *set of all units* is given by $\widetilde{U}$. Each unit U_k (and thus each atomic unit $u_i \in U_k$) is physically connected to a grid node of a power grid G as determined by a *grid node assignment* $gn : \widetilde{U} \to V_G$, with $gn(U_k) = v_{G,l}$. If it holds that $|U| = 1$, U is also referred to as *singleton unit*.

An atomic unit is able to produce or consume electrical energy which is fed into or taken from a power grid. Examples for atomic units are wind energy plants, refrigerators, or pumped storage plants. A unit, in turn, is an aggregation of one or more atomic units which can be of the same or a different type. While this aggregation represents a mere technical pooling of atomic units, an operational aggregation is realized by agents which act as representatives for units in the market.

Definition 3.4: Agent

Let $A = \{a_1, \ldots, a_n\}$ be a finite and nonempty set of rational[2] *agents* a_i exhibiting intelligent behavior. Each agent a_i has a unique identifier ID_{a_i} and is assigned to a unit as defined by a bijective *unit assignment* $u : A \to \widetilde{U}$, with $u(a_i) = U_j$. As software components, agents may either be installed on an embedded system of a singleton unit or on an external hardware, thus representing a node of an ICT network ICT as determined by a *communication node assignment* $cn_a : A \to V_{ICT}$, with $cn_a(a_i) = v_{ICT,k}$. According to their intended purpose, agents are also referred to as *participants* of an electricity market[3].

As discussed in Section 2.2, an intelligent agent represents an autonomous computer system which is able to act in a reactive and proactive fashion and possesses the capa-

[2] For a detailed definition of rationality, see Definition 3.6.
[3] For a detailed definition of an electricity market, see Definition 3.42.

bility to coordinate its actions with other agents in order to achieve its goal. As in the context of this thesis an agent a_i controls a unit U_k, it can generally be installed on

1. an embedded system of an atomic unit, in which case $cn_a(a_i) = cn_U(u_j)$ and $u(a_i) = \{u_j\} = U_k$, or
2. a separate computer system which is connected to one or more atomic units, in which case $cn_a(a_i) \neq cn_U(u_j)\ \forall u_j \in U_k$ and $u(a_i) = U_k$ with $|U_k| \geq 1$.

In order to schedule and operate its unit, an agent reads and writes control data from and to it according to the pursued goal. Because units may comprise more than one atomic unit, it generally possesses the ability to operate multiple atomic units as a single entity by aggregating their technical capabilities. Thus, from an external point of view non-singleton units appear just like any other atomic unit. This approach allows agents to control both single units like solar power or CHP plants and aggregations of single units like households or wind farms. However, as specified in Definition 3.3, a unit is always connected to a single node of a power grid. A distribution of included atomic units over the grid would contradict the concept of a neighborhood and a coalition as defined later in this section.
Figure 3.3 shows how the previously discussed concepts relate to each other. In particular, it depicts the above described possibilities where an agent can be installed meaning that an ICT node can either be given by an agent *and* an atomic unit (case 1) or an agent *or* an atomic unit (case 2). As a consequence, the topology of the ICT network is typically not equal to the one of the underlying power grid.
With regard to its behavior, an agent acts according to a strategy which determines the choice of its actions in order to solve a given task.

Definition 3.5: Strategy

Let an agent a be located in an *environment Env* being defined as a set of *states* $Env = \{s, s', \ldots\}$ which can be entered over time. Moreover, let $Act = \{\alpha, \alpha', \ldots\}$ be a set of *actions* α which can be executed by a. A *run* r of agent a is then defined as a sequence of alternating states and actions $(s_i, \alpha_i, \ldots)$ which starts with a state s and ends either with a state s or an action α.
Let ϑ be a *task* to be solved by a representing the state $s_\vartheta = \vartheta$ which occurrence implicates its fulfillment. A *strategy* $str_\vartheta \in Act^j$[4] for the solution of ϑ is then defined as a sequence of actions $str_\vartheta = (\alpha_1, \ldots, \alpha_j)$ which is the response to a sequence of environmental states $(s_1, \ldots, s_{j+1}) \in Env^{j+1}$, with $s_{j+1} = s_\vartheta$. The run $r_\vartheta = (\alpha_1, s_1, \ldots, \alpha_j, s_\vartheta)$ is termed the *solution* of ϑ.

The above definition of a strategy is based on the concept of a run as described in [Woo09]. In the context of this formalism, the environment of an agent a is modeled as a set of states $\{s, s', \ldots\}$ which can be entered over time. The agent responds to a state s by executing an appropriate action α which results in a sequence of alternating

[4] If S is a set, S^j is the j-ary Cartesian product of S.

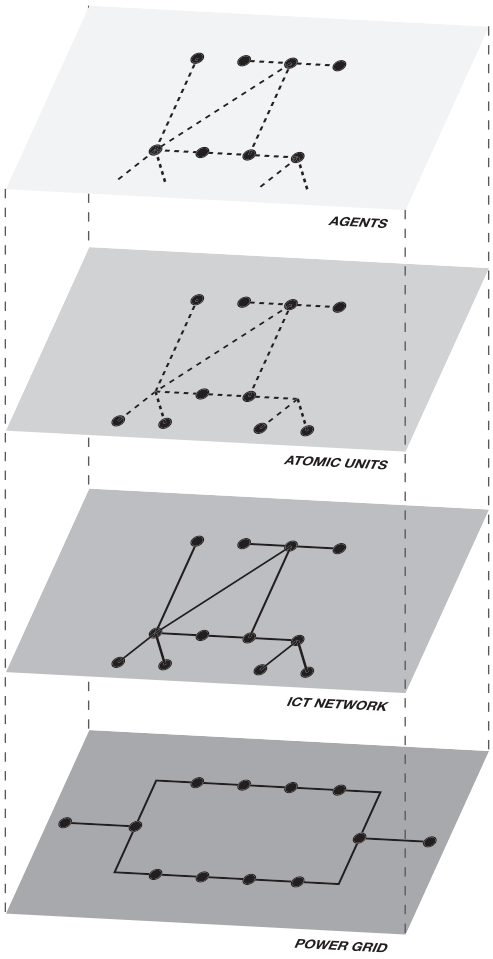


Figure 3.3: Relationships between a power grid, ICT network, units, and agents.

states and actions $(s_i, \alpha_i, \ldots)$. Note that a can generally respond differently to the same environmental state depending on its internal state. The accomplishment of a task ϑ requires the agent to apply an appropriate strategy resulting in a run r_ϑ which final state s_ϑ implicates its fulfillment. As an agent acts as representative for a unit in the market, its strategy determines both its scheduling and trading behavior. It is typically implemented by the designer of the agent (e.g. a plant manufacturer) and configurable by the user with regard to individual preferences.
Which action α is indeed an appropriate response to a given environmental state s is determined by a utility function allowing agents to behave rationally [RN10].

Definition 3.6: Rationality

Given an agent a, let *utility* be defined as a function $util : Env \rightarrow \mathbb{R}$ determining a's benefit if state s is entered. Moreover, let S_α be a random variable which domain is the set of *successor states* $\{s_m, \ldots, s_n\}$, with $n \geq m$, which potentially result from the execution of action α. Finally, let $Pr(S_\alpha = s_i \mid \alpha, e)$ be the *probability of occurrence* of successor state s_i resulting from the execution of α given *evidence* e, i.e. given the information already revealed to agent a. The *expected utility* $EU(\alpha \mid e)$ of an action α given evidence e is then defined as the average utility of all successor states weighted by their probability of occurrence, i.e.

$$EU(\alpha \mid e) = \sum_{i=m}^{n} Pr(S_\alpha = s_i \mid \alpha, e) \cdot util(s_i).$$

Given this, *rationality* is defined as the choice of action α^* maximizing expected utility, i.e.

$$\alpha^* = \arg\max_{\alpha} EU(\alpha \mid e).$$

The above definition of rationality is adopted from decision theory which combines probability theory and utility theory in order to examine rational decision making. Utility function *util* allows an agent to determine which state provides most benefit in general. However, as in stochastic environments actions are not deterministic and potentially lead to different successor states, probabilities of occurrence have to be considered in order to identify the action yielding maximum expected utility. When calculating the expected utility of an action, an agent can take its past experiences into account (we specify this idea in more detail when defining a reasoning scope below). In this context it should be noted that if an agent possesses learning capabilities, its opinion with regard to the benefit of specific states may change over time which in turn has an impact on the calculated utilities and thus its strategy for the accomplishment of its given tasks ϑ.

The approach of always choosing the action which maximizes expected utility is also known as the maximum expected utility (MEU) principle. However, while the latter provides a reasonable and rather appealing way to define rational behavior[5], it is totally silent about how to implement it. Generally, the calculation of utility EU is restricted by the technical capabilities of the system on which the agent is installed, where the complexity of determining probability of occurrence Pr and utility *util* depends on several factors. As already discussed in Section 2.2.1, the environment of an agent can be differently challenging, where particularly in stochastic settings the assessment of the probabilities may require calculation-intensive inference mechanisms. Moreover, in order to calculate the utility of the potential successor states, techniques like searching or planning algorithms have to be applied which computational requirements typically in-

[5] Russel and Norvig even state that the MEU principle can be interpreted as defining all of AI: Creating agents doing the right thing [RN10].

crease with the complexity of the problem. Finally, the difficulty to determine Pr and *util* also depends on the extent to which an agent takes past experiences into account as well as the fact how far it looks ahead.

Definition 3.7: Reasoning Scope

The *reasoning scope* r_{rs} of an agent a with regard to a considered state s_i is defined as a run $r_{rs} = (s_{i-s_p}, \alpha_{i-s_p}, \ldots, s_i, \ldots, \alpha_{i+s_f-1}, s_{i+s_f})$, where $r_{prs} = (s_{i-s_p}, \alpha_{i-s_p}, \ldots, s_{i-1}, \alpha_{i-1})$ is the *past-related reasoning scope* and $r_{frs} = (\alpha_i, s_{i+1}, \ldots, \alpha_{i+s_f-1}, s_{i+s_f})$ is the *future-related reasoning scope* which are taken into account by a when calculating probability of occurrence $Pr(S_\alpha = s_j \mid \alpha, e)$ and utility $util(s_j)$. While $s_p \in \{0, \ldots, i-1\}$ is termed the *past-related extent*, $s_f \in \mathbb{N}_0$ is called the *future-related extent*.

A reasoning scope r_{rs} specifies the information about the past and the future which an agent takes into account when making decisions, where the respective extents are reflected by the values s_p and s_f. More precisely, the past-related reasoning scope r_{prs} represents the past experiences of the agent, i.e. the sequence of environmental states and actions it has gone through. As information which is already revealed to the agent, r_{prs} corresponds to evidence e which it uses when assessing the probability Pr of a potential successor state s. Contrary, the future-related reasoning scope r_{frs} specifies its favoured future course of the world. As this also includes the favoured environmental states, it is taken into account by the agent when determining the utility *util* of a state s. Generally, the utility of a given state s is higher if favoured states s' can be reached from s in the course of ongoing action. When an agent reasons about a task ϑ, the pursued state s_ϑ is part of r_{frs} which also holds for its strategy $str_\vartheta = (\alpha_i, \ldots, \alpha_j)$, where $j \leq (i + s_f - 1)$. When evaluating the expected utility of a cooperation, agents take trust values of their interaction partners into account.

Definition 3.8: Trust

Let T be the *set of all time values*[6]. Given two agents a_i, a_j, *trust* is then defined as a function $trust : T \times A \times A \rightarrow [0, 1]$, where the value $trust((t, a_i, a_j))$ reflects a_i's view of a_j's trustworthiness at time t. The calculation of a trust value is based on *trust criteria* c_{trust} which are considered relevant for its determination. The higher $trust((t, a_i, a_j))$, the higher a_i's trust in a_j.

In the course of a pooling process, the above definition allows an agent a_i to consider the trustworthiness of other participants and decide whether to cooperate with them or not. In this regard, a trust value $trust((t, a_i, a_j))$ reflects its degree of trust in a_j at a specific point in time t. Thus, it allows a_i to reduce its risks by choosing those cooperation partners which meet a minimum level of trustworthiness. The way how trust values are actually calculated depends on the applied trust model, where their assessment is generally based on different criteria c_{trust} which represent trust-related aspects like an agent's

[6] For a detailed definition of the set of all time values, see Definition 3.22.

security measures or its past reliability in the course of product fulfillment. Because trust values lie within the interval [0, 1], they can be compared and exchanged between agents in the course of their interactions, e.g. when negotiating about a cooperation. However, in order to allow for an exchangeability of the values, agents have to use the same trust model and particularly the same trust criteria for their assessment. Otherwise, they make decisions based on different notions.
Before starting a pooling process, it is reasonable for agents to restrict the number of interaction partners. This ability requires an appropriate measure in order to quantify the distance between agents and units.

Definition 3.9: Distance

Given a power grid $G = (V_G, E_G)$, let the *distance* between two grid nodes $v_{G,i}, v_{G,j}$ be defined by a function $dis : V_G \times V_G \rightarrow \mathbb{R}_{\geq 0}$, where $dis((v_{G,i}, v_{G,j}))$ is calculated based on a set of distance weights $dw(v_{G,m}, v_{G,n})$ which relate to power lines $\{v_{G,m}, v_{G,n}\}$ being part of a path between $v_{G,i}$ and $v_{G,j}$. Each distance weight is calculated on the basis of a set of *distance criteria* c_{dis}.
Let the distance between two units U_i, U_j, with $gn(U_i) = v_{G,i}$ and $gn(U_j) = v_{G,j}$, and two agents a_i, a_j, with $u(a_i) = U_i$ and $u(a_j) = U_j$, be defined as the distance between $v_{G,i}$ and $v_{G,j}$.

A distance weight $dw(v_{G,m}, v_{G,n})$ quantifies the physical distance between two adjacent grid nodes $v_{G,m}$ and $v_{G,n}$, where its calculation is based on different distance criteria c_{dis} which are considered relevant for its assessment. Drawing on the definition of a power grid as a weighted graph (see Definition 3.1), the distance between two grid nodes is then determined based on distance weights which are assigned to power lines being part of a path between them. Which weights are actually included in the calculation depends on the applied distance model.
The concept of distance allows agents to form neighborhoods of nearest neighbors.

Definition 3.10: Neighborhood

Given a power grid G and a distance function dis, let the *neighborhood* of a unit U_i be defined as a set of units $N = \{U_1, \ldots, U_{s_N}\} \subseteq \widetilde{U}$ such that

$$dis((gn(U_i), gn(U_1))) \leq \ldots \leq dis((gn(U_i), gn(U_{s_N})))$$

and

$$\forall U_t \in \widetilde{U} \setminus N : dis((gn(U_i), gn(U_t))) \geq dis((gn(U_i), gn(U_{s_N}))).$$

The number of units $s_N \in \mathbb{N}$ defines the *size* of a neighborhood. A neighborhood can be extended by an *extension count* $ext_N \in \mathbb{N}$ specifying the number of units to be added. A neighborhood is termed *extendable* if $|N| < |\widetilde{U}|$.

Correspondingly, let the neighborhood of an agent be defined as the set of agents $N = \{a_1, \ldots, a_{s_N}\} \subseteq A$ which assigned units $u(a_j) = U_j$ comply with the above definition.

While the neighborhood of a unit is given by a set of s_N units constituting its nearest neighbors, the neighborhood of the controlling agent consists of the agents which are assigned to the very same. The concept of a neighborhood allow an agent to restrict the number of interaction partners and thus to reduce communication and computational cost while taking grid-related aspects into account. However, this set is not fixed and can be extended in the course of a pooling process by a defined extension count ext_N in order to increase the probability of a successful cooperation. Note that both size s_N and extension count ext_N are defined in terms of a specific number of neighbors and not a maximum distance threshold. This approach makes a neighborhood independent from the specific characteristics of a power grid in the sense that it can be iteratively extended by a constant number of agents which does not depend on topological aspects, for example.
Within neighborhoods, agents interact in order to cumulate the technical potentials of their units and form coalitions.

Definition 3.11: Coalition

Given a set of agents A, a *coalition* is defined as a finite, nonempty set $C \subseteq A$, where $\widetilde{C} = \mathcal{P}(A) \setminus \emptyset$ is the *set of all coalitions*. The agents $a \in C$ are also referred to as *members* of C. The *size* of a coalition is defined as the number of its members $|C|$, where in case of $|C| = 1$ it is also referred to as *singleton coalition*. Let U_C be the *set of all assigned units*, i.e. $U_C = \{U \mid u^{-1}(U) = a \wedge a \in C\}$. The *scope* of a coalition $s_C \in \mathbb{R}_{\geq 0}$ is defined as the maximum distance between any two units $U_i, U_j \in U_C$, i.e. $s_C = \max_{U_i, U_j \in U_C} dis((gn(U_i), gn(U_j))$. Each coalition has a designated *representative* $a_{rep} \in C$ acting on its behalf in the course of interactions with other actors. In this regard, all members $C \setminus \{a_{rep}\}$ are also referred to as *non-representatives*.

A coalition is an organizational aggregation of one or more agents for the trade of a power product. The organizational binding of each member is temporally restricted in the sense that a coalition dissolves after a product has been physically fulfilled. However, this fact does not hinder agents to form a successful coalition again. As discussed in the context of Definition 3.7, agents are generally able to integrate past experiences into their decision making process which allows for this kind of behavior.
The scope of a coalition C represents the maximum distance which exists between any two units U_i and U_j being assigned to members of C. It thus measures the extent of a coalition in terms of the applied distance function dis. In the course of interactions with other agents, each coalition is represented by a designated member which acts on behalf of the group. Representatives thus constitute organizational interfaces similar to agents which aggregate sets of atomic units and make them appear as singleton units. Besides interacting with other participants, they may also take over further tasks like

the distribution of a final payoff which was gained from a trade. Because the role of a representative can be assumed by any member of a coalition, the designated agent is not critical for its existence.
In the course of a formation process, agents regroup in order to form more valuable coalitions.

Definition 3.12: Regrouping

Let $\widehat{C} \subset \widetilde{C}$. A *regrouping* of two coalitions C_1 and C_2 is defined as a tuple $(\widehat{C}_{old}, \widehat{C}_{new})$, where $\widehat{C}_{old} = \{C_{1,old}, C_{2,old}\}$ is the *set of the original coalitions* $C_1 = C_{1,old}$, $C_2 = C_{2,old}$, and $\widehat{C}_{new} = \{C_{new,1}, \ldots, C_{new,n}\}$ is the *set of the resulting coalitions*.

A regrouping $(\widehat{C}_{old}, \widehat{C}_{new})$ describes a repartitioning of two coalitions C_1 and C_2, where $\widehat{C}_{old}$ represents their original and $\widehat{C}_{new}$ their resulting state. Note that the number of the resulting coalitions may differ from the number of the original ones if the original coalitions either merge or split into more than two new coalitions.
Coalition formation finally results in a partition dividing the set of all agents into disjunct subsets.

Definition 3.13: Coalition Structure

A *coalition structure* CS partitions the set of all agents A into disjunct coalitions, i.e. $\forall C, C' \in CS$ with $C \neq C' : C \cap C' = \emptyset \wedge \bigcup_{C \in CS} C = A$. The *size* of CS is defined as the number of comprised coalitions $|CS|$. The *set of all coalition structures* is given by $\widetilde{CS}$.

The requirement that all coalitions of a coalition structure have to be disjunct guarantees that these are independent from each other with regard to their products. Allowing agents to be part of two or more coalitions at a specific point in time would result in complex dependencies which are hard to handle. For instance, suppose that in the course of coalition formation agent a_1 joins non-singleton coalition C_1 and C_2 and agent a_2 joins non-singleton coalition C_2 and C_3. Further suppose that in the course of the formation process a_1 is asked to increase its amount of electricity contributed to C_1 because of an unforseen event which at the same time requires it to lower its contribution to C_2. If a_1 then reschedules its unit, the members of C_2 are required to react to the new situation which, assuming that a_2 is the only member who has an incentive to adapt its amount, finally requires the members of C_3 to react. As this simple example shows, overlapping coalitions typically result in a high number of interrelated dependencies outweighing potential benefits. Moreover, from a technological point of view, coalitions are particularly formed in order to aggregate the potentials of small-scale units which further splitting is not reasonable.
The ability of a coalition to fulfill its pursued goal is quantified through a dedicated value.

Definition 3.14: Coalition Value

The *coalition value* of a coalition C is defined by a function $v : \widetilde{C} \rightarrow [0, 1]$, where $v(C)$ is also referred as the *worth* of C.

Basically, a coalition value reflects the degree to which its members are able to fulfill the commonly pursued goal. Restricting the codomain of function v to the interval $[0, 1]$ allows for a comparison of different values in the course of a formation process.
Although a higher coalition value typically leads to higher revenues or lower expenses of a coalition, it does not directly correspond to the monetary amount which it finally receives or pays when trading a product.

Definition 3.15: Payoff

Let C be a coalition fulfilling a product p[7]. The *payoff* $\check{\rho}$ resulting from a trade of p is defined as the final total cost which C receives from or pays to its trading partner.

Depending on whether a coalition produces or consumes electrical energy, the payoff can thus be a positive or negative value[8]. After a trade has been successfully completed, it is finally distributed by the coalition members among each other.

Definition 3.16: Payoff Distribution

Given a payoff $\check{\rho}$ of a coalition C, the *payoff distribution* of $\check{\rho}$ is defined by a function $\psi : \widetilde{C} \times \mathbb{R} \rightarrow \mathbb{R}^{|C|}$, where *distribution vector* $\psi((C, \check{\rho})) = \mathbf{x}_C$ specifies the individual *shares* $(x_1, \ldots, x_n)$ of the coalition members $\{a_1, \ldots, a_n\}$. The calculation of $\mathbf{x}_C$ is based on a set of *payoff distribution criteria* c_ψ.

Distribution function ψ distributes a coalition's payoff $\check{\rho}$ among its members, where the shares x_i represent the individual revenues or expenses. The distribution is based on a set of dedicated criteria c_ψ which are considered relevant for its calculation, like a member's contributed amount of electrical energy or its reliability in the course of physical product fulfillment.
As a measure of a coalition's worth, the coalition value forms the basis for the assessment of the value of a coalition structure.

Definition 3.17: Mean Coalition Value

Given a coalition structure CS with $|CS| = n$, the *mean coalition value* is defined as

$$\overline{v(C)} = \frac{\sum_{C \in CS} v(C)}{n}.$$

[7] For a detailed definition of a product, see Definition 3.33.
[8] Note that in game theory the term payoff is often used as synonym for coalition value $v(C)$. However, because of the considered problem, in this thesis we differentiate between both terms as specified above.

As average of all coalition values, the mean coalition value allows agents to optimize the value of the coalition structure. As already described in Section 1.2, in related literature the latter is often evaluated by means of coalition structure value $V(CS)$ which is defined as the sum of all coalition values, i.e. $V(CS) = \sum_{C \in CS} v(C)$. However, in this case the requirement of superadditivity has to be fulfilled, i.e. for all coalitions $C_1, C_2 \in CS$ it must hold that $v(C_1) + v(C_2) \geq v(C_1 \bigcup C_2)$. As this requirement restricts the design of a method for coalition formation, we choose the mean coalition value as global quality measure.
When forming coalitions, agents are able to maximize value with regard to different levels.

Definition 3.18: Value Maximization

According to the applied objective function of the agents, *value maximization* in the context of coalition formation can take place on

- *agent level (a-level)*, in which case each agent a_i maximizes its individual share x_i which it gains as a member of a coalition $C \in CS$ as reflected by objective function

$$\max_{C \in CS} x_{i,C},$$

- *coalition level (C-level)*, in which case the agents maximize the value of single coalitions $C \in CS$ as reflected by objective function

$$\max_{C \in CS} v(C),$$

- *coalition structure level (CS-level)*, in which case the agents maximize the value of coalition structure CS as reflected by objective function

$$\max_{C \in CS} \overline{v(C)}.$$

In terms of Definition 3.5 (strategy) and 3.6 (rationality), an agent's main task ϑ in the course of coalition formation is given by the formation of a coalition which it accomplishes by choosing actions α (and thus a strategy str_ϑ) maximizing expected utility. Which objective function it applies for this task depends on its optimization goal, where it can generally optimize its own payoff, the value of single coalitions, or the value of the whole coalition structure. From a practical point of view, these options correspond to optimizing the value of the unit owner, the value of unit pools, or the value of the whole supply system.
More precisely, on a-level agents strive for an optimization of their individual shares which they gain as a member of a coalition. Consequently, in this case an agent a_i only joins a coalition C with distribution vector $\mathbf{x}_C$ if this action increases its individual payoff x_i. Contrary, on C-level agents optimize the worth of single coalitions C which they join if it raises the coalitions' value $v(C)$. Finally, on CS-level agents maximize the value

of the coalition structure CS in which case they join a coalition C if it improves mean coalition value $\overline{v(C)}$.
Clearly, the goals pursued on the different levels may contradict each other. For instance, if value is maximized on C-level an agent may join a coalition even if it lowers its individual share or the value of the global coalition structure. Which objective function it finally applies depends on the use case as well as the regulations set by the control authorities of the environment. However, assuming that no prescriptions exist, agents are expected to maximize their own payoff as the other two options potentially lead to a lower individual benefit. Thus, if the value of single coalitions or the whole coalition structure is to be optimized, responsible institutions (like federal authorities) have to prescribe appropriate rules which we refer to as regulatory hard constraints (HCs).

Definition 3.19: Regulatory Hard Constraint

A *regulatory hard constraint* hc_a defines a mandatory condition which has to be satisfied by an agent a when it forms coalitions, where the *set of all regulatory hard constraints* is given by HC_a.

Besides prescribing an objective function, regulatory hard constraints may for instance also restrict the maximum size of a coalition in order to prevent market power or require coalitions to include a minimum share of a specific technology (like fuel cells) in order to promote its pervasion.
The fulfillment of a regulatory hard constraint is indicated by a dedicated satisfaction function.

Definition 3.20: Regulatory Hard Constraint Satisfaction

Given an agent a, *regulatory hard constraint satisfaction* with regard to a regulatory hard constraint hc_a is indicated by a function $sat_{hc_a} : A \rightarrow \{0, 1\}$, where

$$sat_{hc_a}(a) = \begin{cases} 1 & \text{if } a \text{ satisfies } hc_a, \\ 0 & \text{else.} \end{cases}$$

While forming coalitions, satisfaction functions sat_{hc_a} allow an agent to validate if it meets all regulatory hard constraints prescribed by its environment.
With regard to its social behavior, an agent may either act egoistic or altruistic.

Definition 3.21: Egoism and Altruism

Let a_k be an agent and ϑ be a task to be fulfilled by a_k given state s_i. Moreover, let $r_{rs} = (s_{i-s_p}, \alpha_{i-s_p}, \ldots, s_i, \alpha_i, \ldots, \alpha_{i+s_f-1}, s_{i+s_f})$, with $0 \leq s_p \leq i-1$ and $0 \leq s_f$, be the reasoning scope of a_k and x_{k,C_1} be the individual share which it gains as a member of coalition C_1. Depending on its social attitude, a_k can then exhibit egoistic or altruistic behavior when choosing action α_i as next step to fulfill ϑ.

Egoism is given if a_k chooses an action such that its individual share x_{k,C_1} is maximal regardless of the impacts on the other agents' shares, the value of any coalition, or the value of the coalition structure, i.e. it holds that

$$\nexists\, C_2 \in CS \setminus C_1 : x_{k,C_2} > x_{k,C_1}.$$

Altruism is given if a_k chooses an action such that its individual share x_{k,C_1} is not maximal provided that it increases the individual share of one or more other agents, the value of a coalition, or the value of the coalition structure, i.e. it holds that

$$\exists\, C_2 \in CS : x_{k,C_2} > x_{k,C_1}.$$

Egoism is termed *conditional* if a_k acts egoistic with regard to a restricted reasoning scope $r'_{rs} = (s_{i-s_p}, \alpha_{i-s_p}, \ldots, s_i, \ldots, \alpha_{j-1}, s_j)$, with $i < j < (i + s_f)$, but altruistic with regard to r_{rs}. Correspondingly, altruism is termed *conditional* if a_k acts altruistic with regard to r'_{rs}, but egoistic with regard to r_{rs}.

The above definition of egoism and altruism is based on an agent's individual payoff which it gains as a member of a coalition. Depending on how payoff is distributed in the course of the overall process, this amount may thus be a definite or an expected value. Figure 3.4 shows how egoism and altruism relate to the concepts of rationality and value maximization as specified in Definition 3.6 and 3.18, respectively. Generally, egoistic or altruistic behavior refers to an action α_i which an agent chooses as a response to a state s_i given its reasoning scope r_{rs} reflecting the past and future information which it takes into account for the decision. If the agent is egoistic, the chosen action always maximizes its individual share x_i regardless of the consequences on the other agents, any coalition, or the coalition structure. It follows that the agent then acts rational (i.e. $\alpha_i = \alpha^* = \arg\max_\alpha EU(\alpha \mid e)$) if its objective function prescribes an optimization on a-level but irrational in the other two cases. This is reasonable as the optimization of a coalition C or the coalition structure CS may result in a lower individual payoff. Contrary, if the agent is altruistic, the chosen action yields a suboptimal share x_i but increases the value of one or more other agents, a coalition, or the coalition structure. While this behavior is rational if the objective function prescribes an optimization on C-level or CS-level, it is obviously irrational if the agent is expected to maximize its own benefits.
However, the above definition does not prevent an egoistic agent from making intermediate altruistic decisions if these are assumed to finally yield a higher individual payoff. This behavior is referred to as conditional egoism as the altruistic decision is based to an expectation about the future course of the world as reflected by future-related reasoning scope r_{frs}. In other words, in case of conditional altruism an agent acts altruistic with regard to a temporarily restricted reasoning scope r'_{rs}, but egoistic in general. Analogously, in case of conditional egoism an agent acts egoistic with regard to a temporarily restricted reasoning scope r'_{rs}, but altruistic in general.
Having covered the first domain of DYCE-FM, we continue with domain 2 which addresses aspects related to the scheduling and operation of a unit.

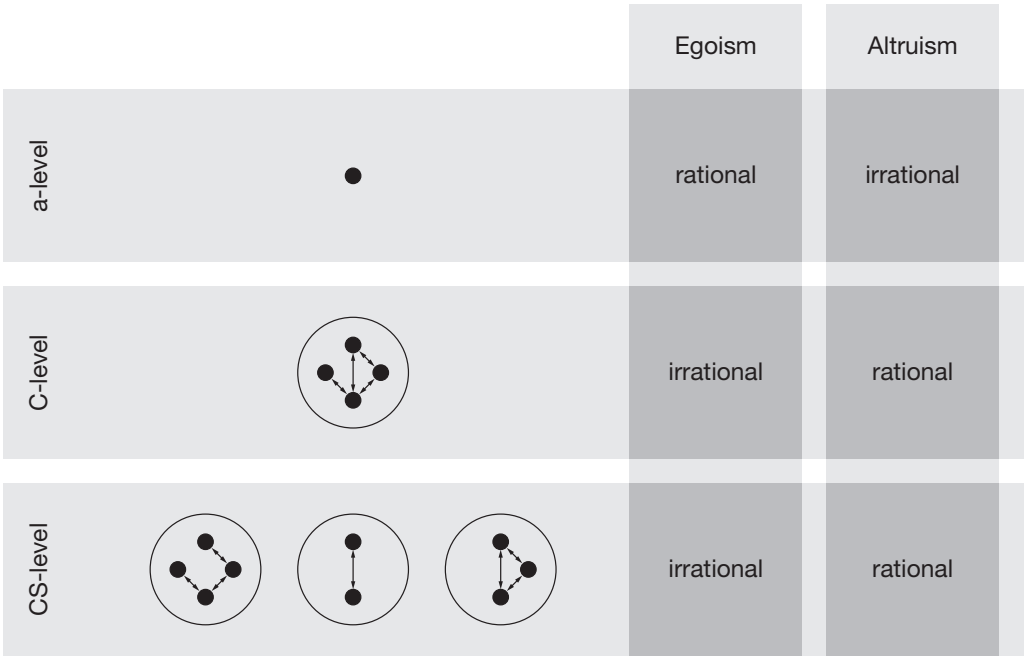


Figure 3.4: Rational egoistic and altruistic behavior in the context of value maximization.

3.2 Domain 2 – Planning Horizon, Operation Schedule

General purpose of a coalition is the fulfillment of a power product through the aggregation of the technical potentials of its members' units. From an agent's point of view, the task of coalition formation is thus directly associated with the scheduling of its units' operation. Generally, all planning activities are done with regard to a planning horizon which constitutes a discretized time frame for which agents plan in advance.

Definition 3.22: Planning Horizon

Let t_{bu} be a *time base unit* and $T \subseteq \mathbb{N}_0$ be the set of all time values measured in t_{bu}. A *planning horizon* is then defined as a set $T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq i_{max}\}$ with $t_{pl}^{(i)} = [t_{i \cdot j}, t_{(i+1) \cdot j})$ being a *planning interval*. Let $t_{max} \in \mathbb{N}$ be the *planning horizon length* and $\Delta t \in \mathbb{N}$ be the *planning interval length* both measured in t_{bu}. Then it holds that

- $t_{i \cdot j} \in T$ is a *time* which is mapped to a *point in real time* t_{rt} by a *time mapping* $\tau : T \to \mathbb{R}, t \mapsto t_{rt}$,
- i is an interval index and i_{max} is the maximum interval index in a planning horizon,

$$i_{max} = \frac{t_{max}}{j \cdot t_u} - 1,$$

where $t_u \in \mathbb{N}$ is a constant, measured in t_{bu}, determining the *time unit* of a planning interval, Δt being a multiple of t_u, and

- j is the interval length measured in the time unit as defined by t_u,

$$j = \frac{\Delta t}{t_u}.$$

A planning horizon is preceded by a *coalition formation period* $t_{CF} = [t_k, t_l]$ consisting of one or more *formation intervals* $t_{f,p_{tmp}} = [t_m, t_n]$, with $t_m \geq t_k$ and $t_n \leq t_l$, in which coalitions with regard to a specific product template[9] p_{tmp} are formed.

Figure 3.5 visualizes the concept of a planning horizon and a product horizon, where the latter is defined later in Definition 3.22. Basically, a planning horizon defines the time interval for which an agent schedules the operation of its unit in advance. As described in more detail below, an operation schedule forms the basis for the contribution of an agent to the product of a coalition. Thus, the temporal resolution of the applied planning horizon has to comply with both the technical characteristics of the controlled unit and the temporal constraints prescribed by the market for the specification of a product. For instance, at the German intraday market the minimum definable time interval is given by a 15-minute period, while at the day-ahead market hour products can be traded as shortest period. Against this background, Definition 3.22 allows for a flexible specification of a planning horizon according to the given requirements. As depicted in Figure 3.5, a horizon T_{pl} is generally made up of a set of successive, equally distant planning intervals. The interval length Δt can be specified to meet the given temporal requirements, where time base unit t_{bu} and time unit t_u allow to determine a reasonable resolution (e.g. minutes or hours). Each planning interval is unambiguously defined through its mapping to a unique time interval in real time. The following examples illustrate how a planning horizon is reasonably specified in practice.

Example 3.1: Planning Horizon

Assume an agent which is participant of an intraday market and schedules the operation of its unit in 15-minute periods on a daily basis. With regard to the applied planning horizon T_{pl}, this corresponds to a time base unit t_{bu} of 1 minute, a time unit t_u of 1 minute, a planning interval length Δt of 15 minutes, and a planning horizon length t_{max} of 1440 minutes (i.e. 1 day) which is divided into 96 planing intervals. Thus, T_{pl} is defined by the settings $\Delta t = 15$, $t_u = 1$, and $t_{max} = 1440$, implicating that $j = 15$ and $T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq 95\}$.

As depicted in Figure 3.5, a planning horizon is preceded by a coalition formation period t_{CF} specifying the time frame in which coalitions are formed for products which are to be physically fulfilled within T_{pl}. Generally, the formation processes are scheduled with

[9] For a detailed definition of a product template, see Definition 3.38.

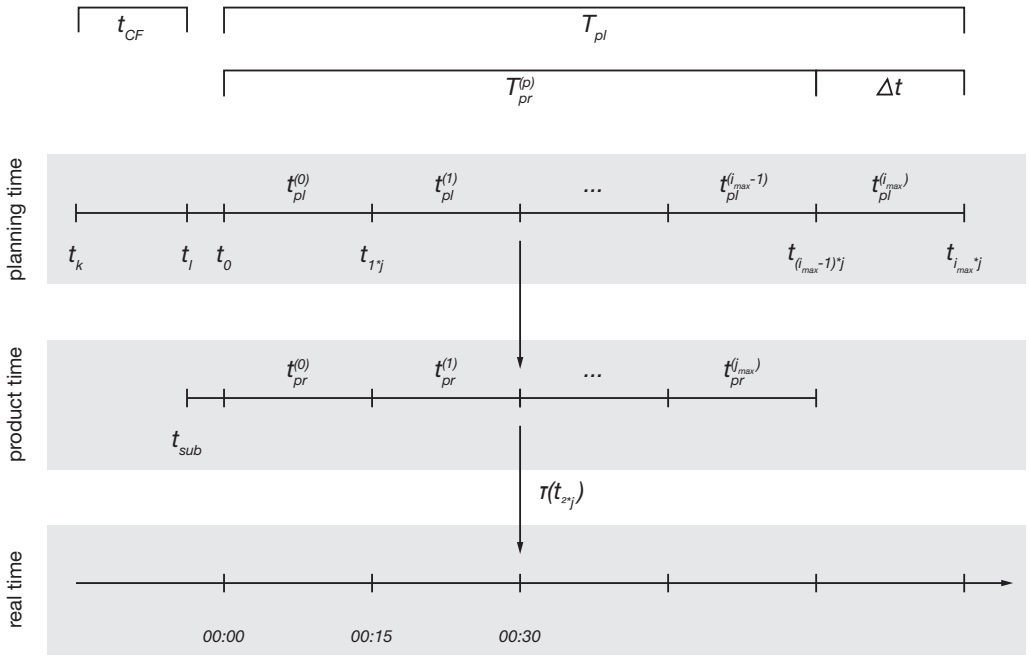


Figure 3.5: Planning horizon and product horizon.

regard to product templates p_{tmp} which formed the basis for the specification of the products. The formation processes for different templates can be arranged in a parallel or a sequential fashion, where t_{CF} may comprise one or more formation intervals $t_{f,p_{tmp}}$. As already mentioned above, planning horizons are used by agents in order to schedule the operation of their units.

Definition 3.23: Operation Schedule

Given a planning horizon $T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq i_{max}\}$, the *operation schedule* of a unit U is defined as a function $os_U : T_{pl} \rightarrow \mathbb{R}^3$, where $os_U(t_{pl}^{(i)}) = (e_U^{(i)}, err_U^{(i)}, c_U^{(i)})$ specifies the *amount of electrical energy* $e_U^{(i)}$ which is produced or consumed by U with *error* $err_U^{(i)}$ at *cost* $c_U^{(i)}$ in planning interval $t_{pl}^{(i)}$. Electricity amount $e_U^{(i)}$ can be equivalently expressed as *amount of electric power*

$$pow_U^{(i)} = \frac{e_U^{(i)}}{\Delta t}$$

which is provided or demanded by U over $t_{pl}^{(i)}$, with Δt being the planning interval length. Let e_{os_U} be the *amount of electrical energy* produced or consumed by U with *error* err_{os_U} at *cost* c_{os_U} in T_{pl}. The *set of all operation schedules* is given by OS_U.

An operation schedule specifies the amount of electrical energy $e_U^{(i)}$ which is produced or consumed by a unit U with a specific error $err_U^{(i)}$ at a specific cost $c_U^{(i)}$ in each planning interval $t_{pl}^{(i)}$ of a planning horizon T_{pl}. Depending on the unit type, the electricity amount can either be a positive or a negative value. Moreover, as the operational behavior of non-controllable units like solar power or wind energy plants may vary within $t_{pl}^{(i)}$, it represents the average amount produced or consumed over the interval. It is typically measured in kWh or MWh and can be equivalently expressed as the power which is provided or demanded by the unit over the planning interval. For instance, an amount of $e_U^{(i)} = 25\,\text{kWh}$ which is produced over an interval of $\Delta t = 15\,\text{min}$ can also be specified as a power volume of $pow_U^{(i)} = 100\,\text{kW} = \frac{25\,\text{kWh}}{0.25\,\text{h}}$. The error $err_U^{(i)}$ and cost $c_U^{(i)}$ are determined based on a dedicated error and cost model, respectively. Throughout coalition formation, an agent is able to make changes to an operation schedule according to its planning activities.
Specific unit types offer flexibilities with regard to their scheduled amounts of electrical energy.

Definition 3.24: Operational Flexibility

Given a planning horizon T_{pl} and an operation schedule os_U, the *operational flexibility of a unit U* in a planning interval $t_{pl}^{(i)}$ is defined as a pair $of_U^{(i)} = (e_{U,-}^{(i)}, e_{U,+}^{(i)}) \in \mathbb{R}^2$ indicating the bounds within which the amount of produced or consumed electrical energy is adjustable, where *negative bound* $e_{U,-}^{(i)}$ and *positive bound* $e_{U,+}^{(i)}$ are specified relative to $e_U^{(i)}$. Operational flexibility $of_U^{(i)}$ is termed *constrained* if an adjustment of $e_U^{(i)}$ in planning interval $t_{pl}^{(i)}$ has consequences on the amount $e_U^{(i-j)}$ or $e_U^{(i+j)}$ in a preceding interval $t_{pl}^{(i-j)}$ or succeeding interval $t_{pl}^{(i+j)}$.
The *operational flexibility of a coalition C* in a planning interval $t_{pl}^{(i)}$ is defined as the sum of the flexibilities of the members' units, i.e.

$$of_C^{(i)} = (e_{C,-}^{(i)}, e_{C,+}^{(i)}), \text{where } e_{C,-}^{(i)} = \sum_{U \in U_C} e_{U,-}^{(i)} \wedge\ e_{C,+}^{(i)} = \sum_{U \in U_C} e_{U,+}^{(i)}.$$

The operational flexibilities of a unit provide some key advantages to the controlling agent when it forms coalitions. First, they allow the agent to choose from a wider range of possibilities when deciding which products to trade or which contributions to make to a coalition. Furthermore, they typically lead to a more reliable product fulfillment as unexpected deviations from scheduled electricity amounts can be flexibly compensated. In the context of coalition formation, operational flexibilities are even more valuable as coalition members can also compensate errors of others which finally prevents a retrieval of expensive control reserve. Formally, an operational flexibility $of_U^{(i)}$ reflects the potential of a unit U to adapt its scheduled amount $e_U^{(i)}$ in planning interval $t_{pl}^{(i)}$. More precisely, it specifies the positive and negative amount of electrical energy which can be added to or subtracted from $e_U^{(i)}$ as defined by the corresponding bounds $e_{U,+}^{(i)}$ and $e_{U,-}^{(i)}$.

However, with regard to operation scheduling in general and coalition formation in particular it is important to differentiate between constrained and unconstrained flexibilities. Making use of constrained flexibilities has an impact on the electricity amount of one or more preceding or succeeding planning intervals and is thus equivalent to a *shift* of production or consumption. Contrary, exploiting unconstrained flexibilities corresponds to an *increment* or *decrement* of production or consumption which concerns a single planning interval only and has no impact on the amounts of other intervals. Similar to overlapping coalitions, in the context of coalition formation constrained flexibilities may lead to an operational rescheduling of other units as exemplified in the following.

Example 3.2: Constrained Operational Flexibility

Assume an agent a which controls a cold store U with a demand of 100 kW and a cooling period of $k \cdot \Delta t$. Because the unit does not cool constantly, it is able to delay its demand within specific temporal bounds as long as the given temperature-related constraints are met. I.e., in a cooling period it potentially provides an operational flexibility of $of_U^{(i)} = (0\,\text{kWh}, 25\,\text{kWh})$, and a flexibility of $of_U^{(i)} = (-25\,\text{kWh}, 0\,\text{kWh})$ otherwise. Now suppose that a is member of a coalition C_1 to which it contributes an amount of -25 kWh in planning interval $t_{pl}^{(j)}$. If a then shifts its demand from $t_{pl}^{(j)}$ to $t_{pl}^{(j+1)}$, in $t_{pl}^{(j+1+k)}$ its unit consumes an amount of -25 kWh instead of 0 kWh. Given that in this planning interval a is member of another coalition C_2, the new deficit of -25 kWh has to be compensated by other participants of C_2.

As the above example shows, constrained flexibilities can lead to an ongoing operational rescheduling of units requiring high coordination efforts between agents. They have thus to be appropriately taken into account by a method for coalition formation in order to prevent that the resulting complexity outweighs the benefits.
The varying operational characteristics of units, as reflected by their operation schedules and operational flexibilities, can be used in order to define classes of unit types.

Definition 3.25: Unit Type

The *unit type ut* of a unit U is defined by its capability to produce or consume amounts of electrical energy $e_U^{(i)}$ as well as its operational flexibility $of_U^{(i)} = (e_{U,-}^{(i)}, e_{U,+}^{(i)})$. It can be one of the following categories:

- *inflexible producer* ut_{ip}, for which generally holds that

$$e_U^{(i)} \in \mathbb{R}_{\geq 0} \qquad \wedge \qquad e_{U,-}^{(i)} = 0, e_{U,+}^{(i)} = 0,$$

- *flexible producer* ut_{fp}, for which generally holds that

$$e_U^{(i)} \in \mathbb{R}_{\geq 0} \qquad \wedge \qquad e_{U,-}^{(i)} \leq 0, e_{U,+}^{(i)} \geq 0,$$

- *inflexible consumer* ut_{ic}, for which generally holds that

$$e_U^{(i)} \in \mathbb{R}_{\leq 0} \qquad \wedge \qquad e_{U,-}^{(i)} = 0, e_{U,+}^{(i)} = 0,$$

- *flexible consumer* ut_{fc}, for which generally holds that

$$e_U^{(i)} \in \mathbb{R}_{\leq 0} \qquad \wedge \qquad e_{U,-}^{(i)} \leq 0, e_{U,+}^{(i)} \geq 0,$$

- *storage*, ut_s, for which generally holds that

$$e_U^{(i)} \in \mathbb{R} \qquad \wedge \qquad e_{U,-}^{(i)} \leq 0, e_{U,+}^{(i)} \geq 0.$$

Inflexible and flexible producers are also referred to as *producers* ut_p, while inflexible and flexible consumers are also referred to as *consumers* ut_c. Moreover, flexible producers, flexible consumers and storage are also termed *flexible* units, while inflexible producers and inflexible consumers are also termed *inflexible* units.

The above categorization of unit types is based on those operational characteristics which are important for the formation of coalitions and the fulfillment of power products. With regard to exemplary technologies, the different types can be described as follows:

- *Inflexible producers* are units like solar power or wind energy plants which feed electrical energy into the power grid according to the availability of the energy source. Because of their lack of operational flexibility, controlling agents can not dynamically respond to changing conditions and are only able to contribute fixed electricity amounts to a coalition.

- *Flexible producers* are units like CHP plants which feed electricity into the grid and are flexible with regard to their generation. Controlling agents are thus able to dynamically respond to occurring changes and contribute electricity amounts in a more reliable fashion.

- *Inflexible consumers* are units like industrial production machines or dishwashers which take electricity from the grid according to their inflexible demand. As in case of inflexible producers, controlling agents are thus only able to contribute fixed electricity amounts to a coalition.

- *Flexible consumers* are units like refrigerators or electric heating systems which take electricity from the grid but are flexible with regard to their demand. Thus, controlling agents are able to adapt to changing conditions and contribute electricity amounts in a more reliable way.

- *Storage* refers to units like batteries or pumped storage plants which are able to produce and consume electrical energy in a flexible fashion. Thus, controlling agents are able to respond to new conditions and reliably contribute positive and negative electricity amounts to coalitions.

Note that the defined unit types are not restricted to singleton units, i.e. in each planning interval $t_{pl}^{(i)}$ non-singleton units can be assigned to one of the above categories as well according to the operational characteristics of their comprised atomic units (cf. Definition 3.3).
As agents act as representatives for their unit in the market, their market role can be derived from the operational behavior of their unit.

Definition 3.26: Market Role

An agent a takes on the market role of a *producer* or a *consumer* according to the given operation schedule os_U of its assigned unit $u(a) = U$.

Since the market role of an agent is determined by the operation schedule of its controlled unit, it may change over time. More precisely, if an agent is assigned to a storage, it is able to act as a producer or a consumer depending on whether the unit is charging or discharging. The same holds if it controls a non-singleton unit U because the cumulated operational behavior of the comprised atomic units $u \in U$ may vary over time. However, as in case of the unit type, the market role of an agent is unambiguously defined in each planning interval $t_{pl}^{(i)}$.
In the course of operation scheduling, an agent is bound to the operational restrictions of its unit.

Definition 3.27: Operation Hard Constraint

An *operation hard constraint* hc_U is a technical restriction of a unit U with regard to its operation. The *set of all operation hard constraints* is given by HC_U.

Operation hard constraints are requirements which have to be mandatorily met by an agent when creating or updating an operation schedule of its controlled unit. An overview of basic operation constraints is given in Table 3.1, where operation soft constraints are defined in Definition 3.29 below. As indicated, the hard constraints can be further distinguished into unconditional ($hc_{U,1}$-$hc_{U,6}$) and conditional constraints ($hc_{U,7}$-$hc_{U,10}$). While the former are fixed restrictions resulting from the technical characteristics of a unit, the latter may change over time according to a unit's state. For instance, while the maximum power level of a solar power plant never changes, its electricity production varies according to the given weather conditions.
The fulfillment of a operation hard constraint can be verified through a corresponding satisfaction function.

id	operation constraint	ut_{ip}	ut_{fp}	ut_{ic}	ut_{fc}	ut_s
$hc_{U,1}$	do not fall below minimum power	•	•	•	•	•
$hc_{U,2}$	do not exceed maximum power	•	•	•	•	•
$hc_{U,3}$	do not fall below minimum operation time	•	•	•	•	•
$hc_{U,4}$	do not exceed maximum operation time	•	•	•	•	•
$hc_{U,5}$	do not fall below minimum capacity					•
$hc_{U,6}$	do not exceed maximum capacity					•
$hc_{U,7}$	do not exceed feasible maximum power	•	•	•	•	•
$hc_{U,8}$	comply with charging level					•
$hc_{U,9}$	do not fall below negative flexibility bound		•		•	•
$hc_{U,10}$	do not exceed positive flexibility bound		•		•	•
$sc_{U,1}$	maximize operation time		•		•	•
$sc_{U,2}$	minimize changes in power level		•		•	•
$sc_{U,3}$	minimize number of shutdowns	•	•	•	•	•

Table 3.1: Basic operation hard and soft constraints (adapted from [Trö10]).

Definition 3.28: Operation Hard Constraint Satisfaction

Given an operation schedule os_U, *operation hard constraint satisfaction* with regard to an operation hard constraint hc_U is indicated by a function $sat_{hc_U} : OS_U \rightarrow \{0, 1\}$, where

$$sat_{hc_U}(os_U) = \begin{cases} 1 & \text{if } os_U \text{ satisfies } hc_U, \\ 0 & \text{else.} \end{cases}$$

Satisfaction function sat_{hc_U} allows an agent to verify if an operation schedule os_U is compliant with a specific hard constraint hc_u. In order to be feasible, a schedule has to meet all given operation hard constraints HC_U.
In contrast to operation hard constraints, operation soft constraints (SCs) are targets which agents pursue to attain but which are not mandatory.

Definition 3.29: Operation Soft Constraint

An *operation soft constraint* sc_U is an objective with regard to the operation of a unit U which an agent tries to achieve in order to maximize value. The *set of all operation soft constraints* is given by SC_U.

As can be seen from the examples in Table 3.1, operation soft constraints are operational optimization goals which achievement is desirable but not mandatory. For instance, a continuous operation of a CHP plant is economically beneficial but may be neglected if no better schedule can be found. As in case of operation hard constraints, the fulfill-

ment of operation soft constraints can be verified through a corresponding satisfaction function.

Definition 3.30: Operation Soft Constraint Satisfaction

Given an operation schedule os_U, *operation soft constraint satisfaction* with regard to an operation soft constraint sc_U is indicated by a function $sat_{sc_U} : OS_U \rightarrow [0, 1]$, where

- $sat_{sc_U}(os_U) = 0$ if os_U does not satisfy sc_U,
- $sat_{sc_U}(os_U) \in (0, 1)$ if os_U partially satisfies sc_U,
- $sat_{sc_U}(os_U) = 1$ else.

As operation soft constraints do not have to be mandatorily achieved, satisfaction function sat_{sc_U} also indicates a partial fulfillment. In the course of planning, an agent can thus verify to which degree a given operation schedule attains a specific target.
The set of all operation hard constraints defines the set of operation schedules which are generally feasible.

Definition 3.31: Operation Schedule Space

Given the set of all operation schedules OS_U of a unit U, the *operation schedule space* OSS_U of U is defined as the set of operation schedules which satisfy the set of all operation hard constraints HC_U, i.e. $OSS_U = \{os_U \mid sat_{hc_U}(os_U) = 1 \; \forall hc_U \in HC_U\}$.

In order to identify the most beneficial operation schedule which it uses as basis for its contributions to the products of coalitions, an agent has to generate the operation schedule space of its controlled unit comprising all feasible schedules. Because inflexible units are not able to vary their production or consumption, their schedule space always consists of one schedule only. Contrary, the schedule space of flexible units typically comprises large amounts of feasible schedules because of their operational degrees of freedom. In this case, agents have to apply appropriate techniques in order to handle the high demand for resources associated with their generation and maintenance [BS14].
Based on the operation schedule space of its unit, an agent identifies its favoured products for which coalitions to form. Related aspects are covered by the third domain of DYCE-FM as described next.

3.3 Domain 3 – Product Horizon, Product

As specified in Definition 3.22, a planning horizon represents the future time frame for which an agent schedules the operation of its unit in advance. As an operation schedule forms the basis for an agent's contribution to the product of a coalition, a planning horizon also defines the time interval in which products can be physically fulfilled.

Definition 3.32: Product Horizon

Given a planning horizon $T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq i_{max}\}$, a *product horizon* $T_{pr}^{(p)} \subseteq T_{pl}$ of a product p is defined as a set $T_{pr}^{(p)} = \{t_{pr}^{(0)}, \ldots, t_{pr}^{(j_{max})}\}$ with $0 \leq j_{max} \leq i_{max}$, where $t_{pr}^{(j)} \in T_{pr}^{(p)}$ is termed a *product interval*.

As shown in Figure 3.5, a product horizon $T_{pr}^{(p)}$ is a set of product intervals which correspond to the planning intervals of a given planning horizon T_{pl} in which a product p is to be physically fulfilled. In case $T_{pr}^{(p)}$ does not totally cover T_{pl}, it can comprise both adjacent and non-adjacent product intervals. Thus, within a planning horizon one or more products can be fulfilled, where the contributions of an agent are based on the corresponding operation schedule.
In order to further clarify how product horizons relate to planning horizons, the following example describes two use cases in which agents plan short-term and long-term.

Example 3.3: Product Horizon

Assume two agents a_1 and a_2 which are participants of a day-ahead spot market and a futures market, respectively. Let time base unit t_{bu} be 1 minute.

- In the context of the spot market, a_1 schedules the operation of its unit in 1-hour periods on a daily basis. The corresponding planning horizon T_{pl} is specified by the settings $t_u = 60$, $\Delta t = 60$, $t_{max} = 1440$, implicating that $j = 1$ and $T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq 23\}$.

  Let $\tau(t_0) = 12{:}00\,\text{am}$, $\tau(t_1) = 01{:}00\,\text{am}, \ldots$ on a considered date. Exemplary product horizons are given by $T_{pr}^{(p_m)} = \{t_{pr}^{(i)} \mid 11 \leq i \leq 12\}$ for an hour product lasting from 12:00 to 13:00 pm and $T_{pr}^{(p_n)} = \{t_{pr}^{(i)} \mid 0 \leq i \leq 7 \wedge 20 \leq i \leq 23\}$ for an off-peak block product lasting from 12:00 to 08:00 am and from 08:00 to 12:00 pm.

- In the context of the futures market, a_2 schedules the operation of its unit in 1-hour periods on a weekly basis. The corresponding planning horizon T_{pl} is specified by the settings $t_u = 60$, $\Delta t = 60$, $t_{max} = 10080$, implicating that $j = 1$ and $T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq 167\}$.

  Let $\tau(t_0) = 12{:}00\,\text{am}, \tau(t_1) = 01{:}00\,\text{am}, \ldots$ on a considered date. An exemplary product horizon is given by $T_{pr}^{(p)} = \{t_{pr}^{(i)} \mid 24 \leq i \leq 47\}$ for a baseload block product lasting 24 hours from 12:00 am on the second day of the considered week.

A product horizon forms the basis for the definition of a product which is fulfilled by a coalition.

Definition 3.33: Product

Given a product horizon $T_{pr}^{(p)} = \{t_{pr}^{(0)}, \ldots, t_{pr}^{(j_{max})}\}$, a *product* fulfilled by a coalition C is defined as a function $p : T_{pr}^{(p)} \rightarrow \mathbb{R}^3$, where $p(t_{pr}^{(i)}) = (e_p^{(i)}, err_p^{(i)}, c_p^{(i)})$ indicates the *amount of electrical energy* $e_p^{(i)}$ which is produced or consumed by C with *error* $err_p^{(i)}$ at *cost* $c_p^{(i)}$ in product interval $t_{pr}^{(i)}$. Electricity amount $e_p^{(i)}$ can be equivalently expressed as *amount of electric power*

$$pow_p^{(i)} = \frac{e_p^{(i)}}{\Delta t}$$

which is supplied or demanded by C over $t_{pr}^{(i)}$, with Δt being the planning interval length. Let e_p be the *amount of electrical energy* which is produced or consumed by C with *error* err_p at *cost* c_p in $T_{pr}^{(p)}$. A product p is termed a *localized product* if it refers to a restricted region of a power grid $G = (V_G, E_G)$ as defined by a set of grid nodes $V_{G,p} \subset V_G$.
Each product is associated with a *time of product submission* $t_{sub} \in T$ by which its specification has to be transmitted to the trading partner at the latest. The *set of all products* is given by P.

A product p determines the amount of electrical energy $e_p^{(i)}$ which is produced or consumed by a coalition with a specific error $err_p^{(i)}$ at a specific cost $c_p^{(i)}$ in each product interval $t_{pr}^{(i)}$ of the corresponding product horizon $T_{pr}^{(p)}$. Generally, the values are defined in terms of the aggregated values of all coalition members. The specification of a product has to be transferred to the trading partner until a specific submission deadline t_{sub}. Note that while a product determines the contractual conditions of a trade (like a bid which is placed at an exchange), in the course of physical fulfillment coalitions may deviate from the specified values because of forecast errors or unforseen events.
The products which are generally tradable in the context of a market are defined by a set of dedicated constraints.

Definition 3.34: Product Hard Constraint

A *product hard constraint* hc_p is a formal restriction with regard to the specification of a product p. The *set of all product hard constraints* is given by HC_p.

Table 3.2 gives an overview of basic product constraints, where product soft constraints are defined in Definition 3.36. As shown, these specify restrictions with regard to the attributes of a product as well as the time intervals in which it can be physically fulfilled. Typically, in mediated markets (like a power exchange) product hard constraints are defined by the mediating party, while in bilateral markets participants are not bound to predefined restrictions and can thus specify contract conditions freely according to their individual needs.

id	product constraint
$hc_{p,1}$	do not fall below minimum volume
$hc_{p,2}$	do not exceed maximum volume
$hc_{p,3}$	do not fall below minimum price
$hc_{p,4}$	do not exceed maximum price
$hc_{p,5}$	comply with product intervals
$sc_{p,1}$	maximize payoff
$sc_{p,2}$	minimize risk

Table 3.2: Basic product hard and soft constraints.

Similar to regulatory and operation hard constraints, product hard constraints can be verified with regard to fulfillment.

Definition 3.35: Product Hard Constraint Satisfaction

Given a product p, *product hard constraint satisfaction* with regard to a product hard constraint hc_p is indicated by a function $sat_{hc_p} : P \rightarrow \{0, 1\}$, where

$$sat_{hc_p}(p) = \begin{cases} 1 & \text{if } p \text{ satisfies } hc_p, \\ 0 & \text{else.} \end{cases}$$

Satisfaction function sat_{hc_p} allows an agent to validate if a given product p is compliant with a specific hard constraint hc_p which is mandatory for p to be tradable.
Contrary, product soft constraints specify goals which agents pursue but are not obliged to achieve.

Definition 3.36: Product Soft Constraint

A *product soft constraint* sc_p is an objective with regard to the specification of a product p which an agent tries to achieve in order to maximize value. The *set of all product soft constraints* is given by SC_p.

As can be seen from the examples in Table 3.2, product soft constraints define objectives which achievement optimizes the utility gained from a trade. The degree of achievement is indicated by a corresponding satisfaction function.

Definition 3.37: Product Soft Constraint Satisfaction

Given a product p, *product soft constraint satisfaction* with regard to a product soft constraint sc_p is indicated by a function $sat_{sc_p} : P \rightarrow [0, 1]$, where

- $sat_{sc_p}(p) = 0$ if p does not satisfy sc_p,

- $sat_{sc_p}(p) \in (0, 1)$ if p partially satisfies sc_p,
- $sat_{sc_p}(p) = 1$ else.

By means of satisfaction function sat_{sc_p} agents are able to verify how well a product p meets the objective defined by a given soft constraint sc_p.
While the set of product hard constraints HC_p refers to all tradable products in a market, product templates are subsets of constraints which are used by agents for the specification of products.

Definition 3.38: Product Template

Given a planning horizon T_{pl} and the set of all product hard constraints HC_p, a *product template* $p_{tmp} \subseteq HC_p$ is defined as a set of product hard constraints hc_p which are relevant for the specification of a corresponding product p. A product p is *compliant* with a product template p_{tmp} if $\forall hc_p \in p_{tmp} : sat_{hc_p}(p) = 1$.
A *template catalog* $P_{tmp} = \{p_{tmp,1}, \ldots, p_{tmp,n}\}$ is a set of product templates which are taken into account by an agent when optimizing its economic utility. A *template portfolio* $TP = \{p_{tmp,1}, \ldots, p_{tmp,m}\} \subseteq P_{tmp}$ is a selection of product templates which are considered as most beneficial and thus used as basis for the specification of a product portfolio[10]. TP is termed *complete* if $\bigcup T_{pr}^{(p_{tmp,i})} \setminus T_{pl} = \emptyset$ and *consistent* if it is complete and it holds that $\bigcap T_{pr}^{(p_{tmp,i})} = \emptyset$, $1 \leq i \leq m$.

The specification of a product is generally based on a product template p_{tmp} which defines all relevant constraints like the product horizon or the maximum definable amount of electrical energy. For example, a template for a peak load product might be specified as $p_{tmp} = \{hc_{p,1}, hc_{p,2}, hc_{p,3}, hc_{p,4}\}$, where

$$
\begin{aligned}
hc_{p,1}: &\quad T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq 95\},\\
hc_{p,2}: &\quad T_{pr}^{(p)} = \{t_{pr}^{(36)}, \ldots, t_{pr}^{(79)}\},\\
hc_{p,3}: &\quad \forall t_{pr}^{(i)} \in T_{pr}^{(p)} : 0.25\,\text{MWh} \leq e_p^{(i)} \leq 100\,\text{MWh},\\
hc_{p,4}: &\quad \forall t_{pr}^{(i)} \in T_{pr}^{(p)} : -500\,€/\text{MWh} \leq c_p^{(i)} \leq 3000\,€/\text{MWh}.
\end{aligned}
$$

I.e., constraint $hc_{p,1}$ defines the planning horizon which is used in the context of constraint $hc_{p,2}$ in order to specify the product horizon which prescribes a physical fulfillment from 09:00 am to 08:00 pm on the considered day. Furthermore, constraint $hc_{p,3}$ and $hc_{p,4}$ define the bounds for the electricity amount and cost, respectively.
While in markets with standardized products (like exchanges) product templates are typically specified by the respective institutions, in other markets these have to be defined by the agents themselves. A template catalog P_{tmp} comprises all templates which an agent takes into account when optimizing utility, where each template is associated with

[10] For a detailed definition of a product portfolio, see Definition 3.39.

a price prediction which allows to estimate the benefit resulting from a trade of a corresponding product. The most beneficial templates are gathered in a template portfolio TP which satisfies the property of completeness if the product horizons prescribed by the templates completely cover the given planning horizon T_{pl}. While in this case the product horizons are still allowed to overlap, the property of consistency ensures that a template portfolio is complete and all prescribed product horizons are disjunct. Thus, a consistent portfolio is always complete, but not vice versa.
A template portfolio is used as basis for the specification of a product portfolio comprising all products for which an agent intends to form coalitions.

Definition 3.39: Product Portfolio

Given a planning horizon T_{pl} and a template portfolio $TP = \{p_{tmp,1}, \ldots, p_{tmp,n}\}$, a *product portfolio* $PP = \{p_{tar,1}, \ldots, p_{tar,n}\}$ is defined as a set of n template-compliant *target products* p_{tar} which an agent intends to trade in T_{pl} by forming coalitions. A *target product* is defined as a function $p_{tar} : T_{pr}^{(p_{tar})} \rightarrow \mathbb{R}^3$, where $p_{tar}(t_{pr}^{(i)}) = (e_{p_{tar}}^{(i)}, err_{p_{tar}}^{(i)}, c_{p_{tar}}^{(i)})$ specifies the *target amount of electrical energy* $e_{p_{tar}}^{(i)}$ which is to be produced or consumed with *target error* $err_{p_{tar}}^{(i)}$ at *target cost* $c_{p_{tar}}^{(i)}$ in $t_{pr}^{(i)}$. Target electricity amount $e_{p_{tar}}^{(i)}$ can be equivalently expressed as *target amount of electric power*

$$pow_{p_{tar}}^{(i)} = \frac{e_{p_{tar}}^{(i)}}{\Delta t}$$

which is to be provided or demanded over $t_{pr}^{(i)}$, with Δt being the planning interval length. All target values of a target product are associated with corresponding *tolerance bands* as defined by a function $p_{tar,tol} : T_{pr}^{(p_{tar})} \rightarrow \mathbb{R}^2 \times \mathbb{R}^2 \times \mathbb{R}^2$, where

$$p_{tar,tol}(t_{pr}^{(i)}) = ((e_{p_{tar},-}^{(i)}, e_{p_{tar},+}^{(i)}), (err_{p_{tar},-}^{(i)}, err_{p_{tar},+}^{(i)}), (c_{p_{tar},-}^{(i)}, c_{p_{tar},+}^{(i)}))$$

specifies the respective *negative tolerance values* and *positive tolerance values*. Tolerance band $(e_{p_{tar},-}^{(i)}, e_{p_{tar},+}^{(i)})$ can be equivalently expressed as $(pow_{p_{tar},-}^{(i)}, pow_{p_{tar},+}^{(i)})$.
A product portfolio PP is termed *complete* if $\bigcup T_{pr}^{(p_{tar,i})} \setminus T_{pl} = \emptyset$ and *consistent* if it is complete and it holds that $\bigcap T_{pr}^{(p_{tar,i})} = \emptyset$, $1 \leq i \leq n$.

The product portfolio of an agent comprises all target products which it finally intends to trade by forming coalitions with like-minded cooperation partners. It is created on the basis of a template catalogue TP which guarantees a market-compliant specification. All target values of a target product are associated with corresponding tolerance bands within which the values are considered to be matched. This allows coalitions to stop a formation process if they only slightly deviate from the target values as defined by the bands.

Having covered all product-related concepts, in the last domain of DYCE-FM we finally consider how agents specify their contributions and these are aggregated by coalitions in order to fulfill their pursued target products.

3.4 Domain 4 – Contribution, Cumulative Contribution

In order to fulfill their favoured target products, agents form coalitions to which they contribute according to the technical capabilities of their units.

Definition 3.40: Contribution

Given an agent a controlling a unit $u(a) = U$, a *contribution* of a to a product p is defined as a function $con_{a,p} : T_{pr}^{(p)} \rightarrow \mathbb{R}^3$, where $con_{a,p}(t_{pr}^{(i)}) = (e_U^{(i)}, err_U^{(i)}, c_U^{(i)})$ specifies the *amount of electrical energy* $e_U^{(i)}$ which is produced or consumed by U with *error* $err_U^{(i)}$ at *cost* $c_U^{(i)}$ in product interval $t_{pr}^{(i)}$ as determined by its operation schedule $os_U(t_{pl}^{(i)}) = (e_U^{(i)}, err_U^{(i)}, c_U^{(i)})$. Let $CON_{C,p}$ be the *set of all contributions* of the members of a coalition C.

A contribution $con_{a,p}$ specifies the amount of electrical energy $e_U^{(i)}$ which an agent contributes to the product of a coalition with a specific error $err_U^{(i)}$ at a specific cost $c_U^{(i)}$ in each product interval $t_{pr}^{(i)} \in T_{pr}^{(p)}$. Its specification is based on the operation schedule of the controlled unit. In the course of coalition formation, agents aggregate their contributions in order to fulfill the pursued target products.

Definition 3.41: Cumulative Contribution

Given the contributions $CON_{C,p}$ of the members of a coalition C pursuing a target product p_{tar}, the *cumulative contribution* of C with regard to p_{tar} is defined as a function $con_{C,p_{tar}} : T_{pr}^{(p_{tar})} \rightarrow \mathbb{R}^3$, where $con_{C,p_{tar}}(t_{pr}^{(i)}) = (e_C^{(i)}, err_C^{(i)}, c_C^{(i)})$ specifies the *cumulative amount of electrical energy* $e_C^{(i)}$ which is produced or consumed by C with *cumulative error* $err_C^{(i)}$ at *cumulative cost* $c_C^{(i)}$ in product interval $t_{pr}^{(i)}$ based on $CON_{C,p}$. Cumulative electricity amount $e_C^{(i)}$ can be equivalently expressed as *cumulative amount of electric power*

$$pow_C^{(i)} = \frac{e_C^{(i)}}{\Delta t}$$

which is provided or demanded by C over $t_{pr}^{(i)}$, with Δt being the planning interval length. Let e_C be the *cumulative amount of electrical energy* which is produced or consumed by

C with *cumulative error* err_C at *cumulative cost* c_C in $T_{pr}^{(p_{tar})}$. C is *successful* and *fulfills* target product p_{tar} if in each product interval $t_{pr}^{(i)} \in T_{pr}^{(p_{tar})}$ it holds that

$$\begin{aligned} e_C^{(i)} &\in [e_{p_{tar}}^{(i)} - e_{p_{tar},-}^{(i)}, e_{p_{tar}}^{(i)} + e_{p_{tar},+}^{(i)}], \\ err_C^{(i)} &\in [err_{p_{tar}}^{(i)} - err_{p_{tar},-}^{(i)}, err_{p_{tar}}^{(i)} + err_{p_{tar},+}^{(i)}], \\ c_C^{(i)} &\in [c_{p_{tar}}^{(i)} - c_{p_{tar},-}^{(i)}, c_{p_{tar}}^{(i)} + c_{p_{tar},+}^{(i)}]. \end{aligned}$$

In case of a successful fulfillment, the product p which can finally be traded by C is defined by its final cumulative contribution, i.e.

$$\forall t_{pr}^{(i)} \in T_{pr}^{(p)} : (e_p^{(i)}, err_p^{(i)}, c_p^{(i)}) = (e_C^{(i)}, err_C^{(i)}, c_C^{(i)}).$$

The cumulative contribution of a coalition is an aggregation of the individual contributions of its members. In the course of coalition formation, a coalition continues with the formation process until its cumulative values lie within the corresponding tolerance bands of the pursued target product or other termination conditions are met. As there may thus be deviations from the target values even if it is successful, the tradable product is finally given by its cumulative contribution and not the target product itself.
Having specified the four domains of DYCE-FM, we are finally ready to define the context in which coalition formation takes place.

Definition 3.42: Electricity Market

An *electricity market* is defined as a tuple $M = (\{M_{s,1}, \ldots, M_{s,n}\}, G)$, where $M_{s,i} = (mt, HC_p)$ is a *submarket* of *market type*

$$mt \in \{search, board, brokered, dealer, exchange, pool\}$$

prescribing a set of product hard constraints HC_p for product specification. The *place of delivery* is given by a power grid G.

An electricity market consists of one or more submarkets and defines a place of delivery which is given by a power grid G connecting all producers and consumers. Each submarket is of one of the types discussed in Section 2.1.2 and prescribes a set of product hard constraints HC_p which has to be taken into account by agents when specifying target products for which coalitions to form.

symbol	description	def.
$\{v_{G,i}, v_{G,j}\}$	power line of a power grid G	3.1
$\{v_{ICT,i}, v_{ICT,j}\}$	communication line of an ICT network ICT	3.2
α	action of an agent	3.5
α^*	action of an agent maximizing expected utility	3.6
Δt	planning interval length	3.22
ϑ	task of an agent	3.5
$\check{\rho}$	payoff of a coalition C	3.15
$\tau(t)$	time mapping of a time t	3.22
$\psi((C, \check{\rho}))$	distribution of a payoff $\check{\rho}$ of a coalition C	3.16
a	agent	3.4
A	set of agents	3.4
a_{rep}	representative of a coalition	3.11
Act	set of actions an agent can execute	3.5
C	coalition of agents	3.11
$\widetilde{C}$	set of all coalitions	3.11
c_C	cost of a cumulative contribution in $T_{pr}^{(p)}$	3.41
$c_C^{(i)}$	cost of a cumulative contribution in $t_{pr}^{(i)}$	3.41
c_{dis}	distance criterion	3.9
c_{os_U}	cost of an operation schedule os_U in T_{pl}	3.23
$\widehat{C}_{new}$	set of coalitions resulting from a regrouping	3.12
$\widehat{C}_{old}$	set of coalitions before a regrouping	3.12
c_p	cost of a product p in $T_{pr}^{(p)}$	3.33
$c_p^{(i)}$	cost of a product p in $t_{pr}^{(i)}$	3.33
$c_{p_{tar}}^{(i)}$	target cost of a target product p_{tar} in $t_{pr}^{(i)}$	3.39
$c_{p_{tar},-}^{(i)}$	negative tolerance value for a target cost $c_{p_{tar}}^{(i)}$	3.39
$c_{p_{tar},+}^{(i)}$	positive tolerance value for a target cost $c_{p_{tar}}^{(i)}$	3.39
c_ψ	payoff distribution criterion	3.16
c_{trust}	trust criterion	3.8
$c_U^{(i)}$	cost of an operation schedule in $t_{pl}^{(i)}$	3.23
$cn_a(a)$	communication node assignment of an agent a	3.4
$cn_U(U)$	communication node assignment of a unit U	3.3
$con_{a,p}(t_{pr}^{(i)})$	contribution of an agent a in $t_{pr}^{(i)}$	3.40
$CON_{C,p}$	set of all contributions of the members of a coalition C	3.40
$con_{C,p}(t_{pr}^{(i)})$	cumulative contribution of a coalition C in $t_{pr}^{(i)}$	3.41
CS	coalition structure	3.13
$\widetilde{CS}$	set of all coalition structures	3.13
$dis((v_{G,i}, v_{G,j}))$	distance between two grid nodes $v_{G,i}, v_{G,j}$	3.9
$dw(v_{G,i}, v_{G,j})$	distance weight relating to a power line $\{v_{G,i}, v_{G,j}\}$	3.9
e	evidence	3.6

e_C	electricity amount of a cumulative contribution in $T_{pr}^{(p)}$	3.41
$e_C^{(i)}$	electricity amount of a cumulative contribution in $t_{pr}^{(i)}$	3.41
E_G	power lines of a power grid G	3.1
E_{ICT}	communication lines of an ICT network ICT	3.2
e_{os_U}	electricity amount of an operation schedule os_U in T_{pl}	3.23
e_p	electricity amount of a product p in $T_{pr}^{(p)}$	3.33
$e_p^{(i)}$	electricity amount of a product p in $t_{pr}^{(i)}$	3.33
$e_{p_{tar}}^{(i)}$	target electricity amount of a target product p_{tar} in $t_{pr}^{(i)}$	3.39
$e_{p_{tar},-}^{(i)}$	negative tolerance value for target electricity amount $e_{p_{tar}}^{(i)}$	3.41
$e_{p_{tar},+}^{(i)}$	positive tolerance value for target electricity amount $e_{p_{tar}}^{(i)}$	3.41
$e_U^{(i)}$	electricity amount of an operation schedule in $t_{pl}^{(i)}$	3.23
$e_{U,-}^{(i)}$	negative operational flexibility bound of a unit U in $t_{pl}^{(i)}$	3.24
$e_{U,+}^{(i)}$	positive operational flexibility bound of a unit U in $t_{pl}^{(i)}$	3.24
Env	environment of an agent	3.5
err_C	error of a cumulative contribution in $T_{pr}^{(p)}$	3.41
$err_C^{(i)}$	error of a cumulative contribution in $t_{pr}^{(i)}$	3.41
err_{os_U}	error of an operation schedule os_U in T_{pl}	3.23
err_p	error of a product p in $T_{pr}^{(p)}$	3.33
$err_p^{(i)}$	error of a product p in $t_{pr}^{(i)}$	3.33
$err_{p_{tar}}^{(i)}$	target error of a target product p_{tar} in $t_{pr}^{(i)}$	3.39
$err_{p_{tar},-}^{(i)}$	negative tolerance value for target error $err_{p_{tar}}^{(i)}$	3.41
$err_{p_{tar},+}^{(i)}$	positive tolerance value for target error $err_{p_{tar}}^{(i)}$	3.41
$err_U^{(i)}$	error of an operation schedule in $t_{pl}^{(i)}$	3.23
$EU(\alpha \mid e)$	expected utility of an action α given evidence e	3.6
ext_N	extension count relating to a neighborhood N	3.10
G	power grid	3.1
$gn(U)$	grid node assignment of a unit U	3.3
hc_a	regulatory hard constraint relevant for an agent a	3.19
HC_a	set of all regulatory hard constraints relevant for an agent a	3.19
hc_p	product hard constraint relating to a product p	3.34
HC_p	set of all product hard constraints relating to a product p	3.34
hc_U	operation hard constraint of a unit U	3.27
HC_U	set of all operation hard constraints of a unit U	3.27
ID_a	unique identifier of an agent a	3.4
ICT	ICT network	3.2
M	electricity market	3.42
M_s	submarket	3.42
mt	market type	3.42
N	neighborhood of a unit	3.10
$of_C^{(i)}$	operational flexibility of a coalition C in $t_{pl}^{(i)}$	3.24

$of_U^{(i)}$	operational flexibility of a unit U in $t_{pl}^{(i)}$	3.24
OS_U	set of all operation schedules of a unit U	3.23
$os_U(t_{pl}^{(i)})$	value of an operation schedule os_U of a unit U in $t_{pl}^{(i)}$	3.23
OSS_U	operation schedule space of a unit U	3.31
P	set of all products	3.33
$p(t_{pr}^{(i)})$	value of a product p in $t_{pr}^{(i)}$	3.33
$p_{tar}(t_{pr}^{(i)})$	value of a target product p_{tar} in $t_{pr}^{(i)}$	3.39
$p_{tar,tol}(t_{pr}^{(i)})$	tolerance bands of a target product in $t_{pr}^{(i)}$	3.39
p_{tmp}	product template	3.38
P_{tmp}	template catalog	3.38
$pow_C^{(i)}$	power amount of a cumulative contribution in $t_{pr}^{(i)}$	3.41
$pow_p^{(i)}$	power amount of a product p in $t_{pr}^{(i)}$	3.33
$pow_{p_{tar}}^{(i)}$	target power amount of a target product p_{tar} in $t_{pr}^{(i)}$	3.39
$pow_{p_{tar},-}^{(i)}$	negative tolerance value for target power $pow_{p_{tar}}^{(i)}$	3.39
$pow_{p_{tar},+}^{(i)}$	positive tolerance value for target power $pow_{p_{tar}}^{(i)}$	3.39
$pow_U^{(i)}$	power amount of an operation schedule in $t_{pl}^{(i)}$	3.23
PP	product portfolio	3.39
$Pr(S_\alpha = s_i \mid \alpha, e)$	probability of occurrence of a state s_i resulting from applying an action α given evidence e	3.6
$qw(v_{ICT,i}, v_{ICT,j})$	quality-weight of a communication line $\{v_{ICT,i}, v_{ICT,j}\}$	3.2
r	run of an agent	3.5
r_ϑ	solution of a task ϑ	3.5
r_{frs}	future-related reasoning scope of an agent	3.7
r_{prs}	past-related reasoning scope of an agent	3.7
r_{rs}	reasoning scope of an agent	3.7
s	state of an environment	3.5
S_α	random variable determining the states potentially resulting from an action α	3.6
s_ϑ	state implicating the fulfillment of a task ϑ	3.5
s_f	future-related extent of a reasoning scope	3.7
s_C	scope of a coalition C	3.11
s_N	size of a neighborhood N	3.10
s_p	past-related extent of a reasoning scope	3.7
$sat_{hc_a}(a)$	regulatory hard constraint satisfaction w.r.t. a	3.20
$sat_{hc_p}(p)$	product hard constraint satisfaction w.r.t. p	3.35
$sat_{hc_U}(os_U)$	operation hard constraint satisfaction w.r.t. os_U	3.28
$sat_{sc_p}(p)$	product soft constraint satisfaction w.r.t. p	3.37
$sat_{sc_U}(os_U)$	operation soft constraint satisfaction w.r.t. os_U	3.30
sc_p	product soft constraint relevant for a product p	3.36
SC_p	set of all product soft constraints relevant for a product p	3.36
sc_U	operation soft constraint relevant for a unit U	3.29
SC_U	set of all operation soft constraints relevant for a unit U	3.29

str_ϑ	strategy of an agent for solving a task ϑ	3.5
t	time value	3.22
T	set of all time values	3.22
t_{bu}	time base unit	3.22
t_{CF}	coalition formation period	3.22
$t_{f,p_{tmp}}$	formation interval	3.22
t_{max}	planning horizon length	3.22
T_{pl}	planning horizon	3.22
$t_{pl}^{(i)}$	planning interval	3.22
$t_{pr}^{(j)}$	product interval	3.32
$T_{pr}^{(p)}$	product horizon of a product p	3.32
t_{rt}	point in real time	3.22
t_{sub}	time of product submission	3.33
t_u	time unit of a planning interval	3.22
TP	template portfolio	3.38
$trust((a_i, a_j))$	trust of an agent a_i in an agent a_j	3.8
u	atomic unit	3.3
U	unit	3.3
$\widetilde{U}$	set of all units	3.3
U_a	set of atomic units	3.3
U_C	set of all units assigned to the members of a coalition C	3.11
$u(a)$	unit assignment of an agent a	3.4
ut	unit type	3.25
ut_c	unit type consumer	3.25
ut_{fc}	unit type flexible consumer	3.25
ut_{fp}	unit type flexible producer	3.25
ut_{ic}	unit type inflexible consumer	3.25
ut_{ip}	unit type inflexible producer	3.25
ut_p	unit type producer	3.25
ut_s	unit type storage	3.25
$util(s)$	utility of a state s	3.6
$v(C)$	coalition value of a coalition C	3.14
$\overline{v(C)}$	mean coalition value of a coalition structure CS	3.17
v_G	grid node of a power grid G	3.1
V_G	grid nodes of a power grid G	3.1
v_{ICT}	communication node of an ICT network ICT	3.2
V_{ICT}	communication nodes of an ICT network ICT	3.2
$\mathbf{x}_C$	payoff distribution vector of a coalition C	3.16
x_i	individual share of an agent a_i	3.16

Table 3.3: Symbols specified in Definition 3.1-3.42.

4 A Method for Dynamic Coalition Formation in Electricity Markets

Having precisely specified the considered problem, in the following we provide a solution to the very same by describing an agent-based self-organization approach for the formation of product-related coalitions termed DYCE – DYnamic Coalition formation in Electricity markets. The proposed method comprises four main activities which are carried out by an agent in an iterative and partially parallel fashion as depicted by the UML diagram in Figure 4.1[1]. Each activity addresses a self-contained subproblem of the overall problem which is associated with a corresponding optimization problem and thus a dedicated computational complexity. The four main activities can be summarized as follows:

Product portfolio management In the course of the first DYCE activity an agent mainly performs three tasks. First, it builds a product portfolio comprising all target products which it intends to trade within its considered planning horizon. The portfolio is created on the basis of a set of product templates for which the agent assumes corresponding price predictions. Second, it chooses an operation schedule from the operation schedule space of its controlled unit which forms the basis for its contributions to coalitions in the course of the following formation processes. Both actions are interrelated in the sense that in conjunction the identified portfolio and schedule are intended to optimize local utility. Finally, the agent plans the coalition formation processes related to its identified target products and then conducts the following three activities according to the created formation schedule.

Neighborhood formation If throughout the upcoming coalition formation an agent initiates negotiations, it first builds a neighborhood of nearest neighbors based on a distance function quantifying the physical distance between the units in the grid. By limiting the set of all agents to a manageable size, neighborhoods allow to reduce communication and computational cost while taking grid-related aspects into account. If a formation attempt with current neighbors was unsuccessful, agents are able to extent their neighborhoods in order to include a wider range of potential cooperation partners.

Coalition formation In the course of the third DYCE activity agents form coalitions within the previously built neighborhoods in order to cooperatively fulfill their initially specified target products. Generally, the formation processes optimize the global value of the whole coalition structure (cf. Definition 3.18). If the formation of a coalition with current neighbors can not be achieved, an agent has generally two options to proceed. First, if its neighborhood is still extendable, it can expand the very same in order to increase the number of potential cooperation partners and thus the probability to form a successful coalition (1). Second, if the neighborhood

[1] The guard conditions of the control flows are specified in detail in the following sections.

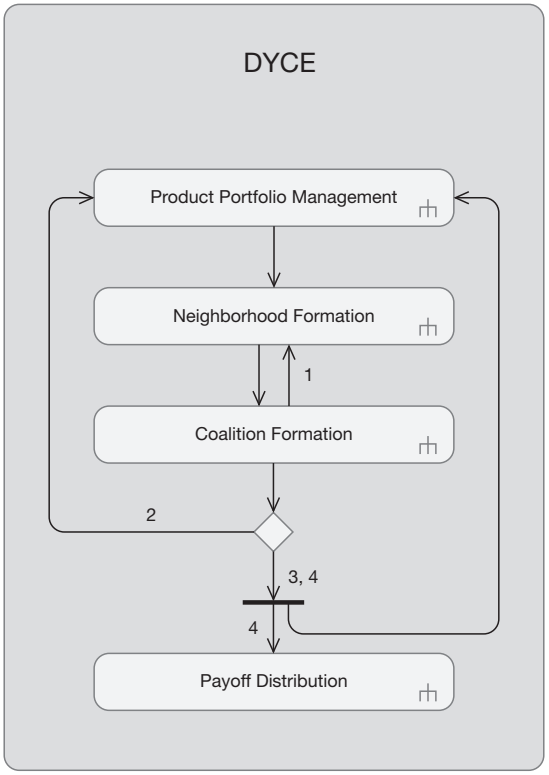


Figure 4.1: DYCE activity diagram.

already covers the whole grid, it can try to specify a new product portfolio comprising a different set of target products (2). However, if both of these options are not feasible, an agent stops the formation process as member of an unsuccessful coalition and schedules the next target product for which a coalition to form (3). The latter also holds if it stops the process as member of a successful coalition, where in this case it also conducts the last activity of the overall process (4).

Payoff distribution If a coalition successfully formed, it finally strives for a trade of its fulfilled target product and a distribution of the resulting payoff. Generally, the applied distribution model is based on a game-theoretical model which enables a fair[2] division of the coalition's utility in terms of the Shapley value. The resulting distribution vector finally specifies the individual shares which the members gain from cooperation.

[2] Because of the computational complexity of the related problem, the model allows for the application of a heuristic in order to efficiently approximate a fair distribution if the size of a coalition gets too large. However, in the following we simply speak of a fair distribution including the case of an approximated solution.

As can be seen from the above summary, DYCE optimizes both local and global utility by taking account of the agents' individual benefits in the course of product portfolio management and optimizing the value of the coalition structure in the context of coalition formation. This combined approach makes the method applicable to a wide range of use cases falling into either of the following two categories:

Application on local system level If applied on local level of the power supply system, DYCE can be used for the control of a restricted set of units which are typically owned by a single party and distributed over a limited region of the grid. For instance, the approach might be applied by an operator of several generators connected to the distribution grid in order to control the units in the sense of a dynamic virtual power plant. In this regard, DYCE can also be used in order to exchange currently applied centralized management systems by a decentralized control.

Application on global system level Alternatively, DYCE can be applied for a decentralized power supply on global system level in order to replace the current approach of a hierarchically organized provision. In this scenario, all planning activities are conducted by the market participants themselves without depending on other service providers, where centralized markets may or may not be part of the system.

Clearly, an application of DYCE on global level is not implementable from scratch and requires an appropriate migration path towards a fully decentralized supply. This can be achieved through a successive deployment in the sense of the first use case which allows to handle integration efforts and reduce the risks associated with the introduction of new technologies.
In what follows, we provide a detailed description of DYCE by covering the above outlined activities in Section 4.1-4.4. In each section, we first discuss the considered problem along with its computational complexity and then detail the developed concepts for its solution. Throughout the descriptions, we make assumptions with regard to different design-related aspects which are as follows:

- A_1: With respect to the communication between agents, we suppose that the underlying ICT network ICT provides perfect transmission quality, i.e. $\forall v_{ICT,i}, v_{ICT,j} \in V_{ICT}$ with $i \neq j : qw(v_{ICT,i}, v_{ICT,j}) = 1$. This particularly includes the assumption that sent messages are transmitted without error and arrive in the specified order. Concepts for the handling of unreliable communication are left to future work (see Section 6.2).

- A_2: We account for the aspect of trust by applying the abstract notion specified in Definition 3.8 while assuming the underlying trust model as given. In the context of evaluation in Chapter 5, we thus further suppose that all participating agents are trustworthy and do not misbehave in the course of the interactions. The design of a comprehensive trust model and the investigation of corresponding threat scenarios are left to future work (see Section 6.2).

- A_3: With regard to a product p, we assume that the amounts of electrical energy $e_U^{(i)}$ which are specified within the corresponding product horizon $T_{pr}^{(p)}$ are either all

positive or all negative, i.e. $\forall t_{pr}^{(i)} \in T_{pr}^{(p)} : e_U^{(i)} > 0$ or $\forall t_{pr}^{(i)} \in T_{pr}^{(p)} : e_U^{(i)} < 0$. With regard to coalition formation, this particularly means that a storage acts either as producer or consumer as a charge and discharge is always realized by means of two different products. Moreover, supply and demand is never matched within coalitions but only through the market.

In order to facilitate discussion in the course of the following descriptions, we sometimes refer to an exemplary use case in which a flexible producer participates in the spot market of the European Energy Exchange and strives for the formation of coalitions a day before order book closure. With regard to the applied planning horizon T_{pl}, we thus assume that $t_{bu} = 1$, $t_u = 1$, $\Delta t = 15$, and $t_{max} = 1440$ (one day), implicating that $j = 15$ and $T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq 95\}$ (cf. Definition 3.22).

4.1 Product Portfolio Management

In the course of the first DYCE activity, an agent initially specifies the different products which it intends to trade within its considered planning horizon and schedules the processes for the formation of corresponding coalitions in order to cooperatively fulfill the very same. Moreover, it identifies an operation schedule which it uses as basis for its contributions throughout the formation processes. In the following sections, we first analyze the optimization problem related to these tasks and then provide appropriate concepts for its solution. In particular, we describe a heuristic which allows to efficiently approximate a theoretical optimum and integrate multiple markets of the same or a different type into the decision making process. The proposed algorithm is independent from the unit type and can thus be applied by arbitrary agents.

4.1.1 Problem Specification and Computational Complexity

The first activity which an agent carries out in the course of the overall process comprises three main tasks two of which are highly interrelated with regard to their goal. First, the agent has to create a product portfolio PP including all target products p_{tar} which it intends to trade within its considered planning horizon T_{pl} in one or more submarkets M_s. The specification of the product portfolio is based on a template portfolio TP which is a selection of product templates p_{tmp} from a template catalog P_{tmp} maximizing the agent's expected utility (for an example of a product template see Section 3.3). As a result, PP finally maximizes the agent's expected utility as well. The choice of which product templates to include in TP is based on the one hand on corresponding interval-related price predictions $c_{p_{tmp}}^{(i)}$ which are the cost which an agent expects with regard to a trade of an accordingly specified target product p_{tar}, with $C_{P_{tmp}}$ being the set of predictions for all templates $p_{tmp} \in P_{tmp}$. On the other hand, it depends on the operation schedule os_U which the agent uses as basis for its contributions to coalitions in the course of formation processes. For instance, for a producer the specification of a target product on the basis of a product template with a high price prediction would be economically unprofitable if its unit can not provide electrical energy in the product horizon prescribed by

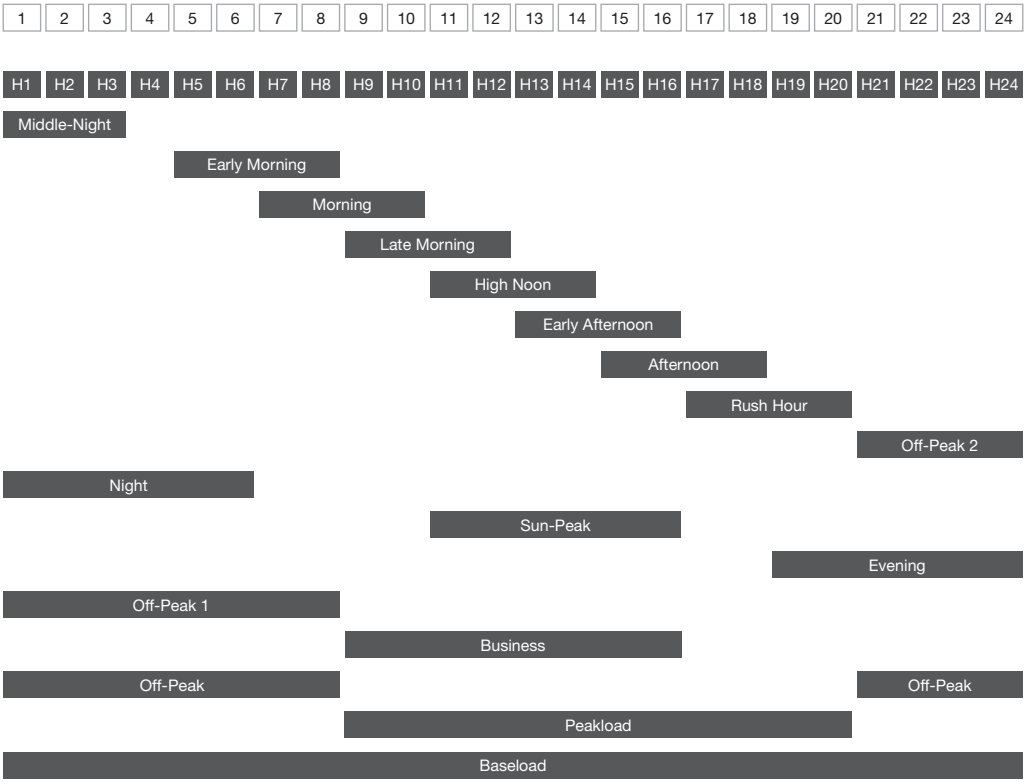


Figure 4.2: Template catalog of the EPEX SPOT day-ahead auction (hour 1-24).

the template. Hence, as second task of the activity, an agent has to identify an operation schedule os_U from the operation schedule space OSS_U of its unit U which is compliant with the product portfolio in the sense that in conjunction both maximize its expected utility. The identified product portfolio and operation schedule are then used as basis for the following coalition formation processes, where the latter have to be finally scheduled by an agent in coordination with other participants.

In order to illustrate the problem of product portfolio management, we refer to our previously described use case in which a flexible producer participates in the EPEX SPOT market. Figure 4.2 gives an overview of the corresponding template catalog P_{tmp} comprising the forty-one product templates p_{tmp} which can generally be used by participants in order to specify their target products. Supposing that the producer assumes different price predictions for the templates (which is typically the case) and the operation schedule space of its controlled unit comprises a large set of feasible schedules, the task of the agent is then to identify a consistent template portfolio TP and schedule os_U which in conjunction maximize its expected utility. Say such a portfolio is given by $TP = \{p_{tmp,Night}, p_{tmp,Morning}, p_{tmp,High\ Noon}, p_{tmp,Afternoon}, p_{tmp,Evening}\}$ covering the whole 24 hours of the considered planning horizon. The agent then uses this set of prod-

uct templates in order to define its final product portfolio PP which might for instance comprise a target product $p_{tar,Afternoon}$ specifying a target electricity amount of 100 kWh being provided with an estimated error of 2 kWh at cost 0.1 €/kWh from 03:00 pm to 06:00 pm on the considered day. Finally, the agent schedules the formation processes related to the five target products within coalition formation period t_{CF} which temporally precedes planning horizon T_{pl} (cf. Definition 3.22).
According to the above problem specification, the activity of product portfolio management is closely related to the task of utility assessment for a considered pair of template portfolio and operation schedule. However, at this stage of the process an agent is yet not able to calculate its final share of the payoff resulting from the trade of a product as determined by the approach applied in the course of the last DYCE activity (see Section 4.4.2). More precisely, it has yet no information about the final coalitions and the members' contributions to their cumulative contributions which means that it can not calculate the corresponding divisions. Hence, at this stage of the process, an agent applies an alternative approach for estimating the benefit resulting from a potential trade: In order to asses the utility which is associated with a pair of a template portfolio TP and operation schedule os_U, it considers the expected surplus which results from the set of template-compliant target products p_{tar} being specified on the basis of os_U. The applied notion of utility thus differs according to the unit type ut of the controlled unit U. More precisely, for a single product template p_{tmp} prescribing a product horizon $T_{pr}^{(p)} = \{t_{pr}^{(0)}, \ldots, t_{pr}^{(j_{max})}\}$ and an operation schedule os_U, with $os_U(t_{pl}^{(i)}) = (e_U^{(i)}, err_U^{(i)}, c_U^{(i)})$, the expected utility of a producer ut_p and consumer ut_c is defined as

$$utility(p_{tmp}, os_U) = \begin{cases} \sum_{i=0}^{j_{max}} e_U^{(i)} \cdot c_{p_{tmp}}^{(i)} - e_U^{(i)} \cdot c_U^{(i)} & \text{if } U \text{ is of type } ut_p, \\ \sum_{i=0}^{j_{max}} e_U^{(i)} \cdot c_U^{(i)} - e_U^{(i)} \cdot c_{p_{tmp}}^{(i)} & \text{if } U \text{ is of type } ut_c. \end{cases} \tag{4.1}$$

While for a producer utility is the profit which it obtains from the sale of a target product being specified on the basis of the given product template p_{tmp} and schedule os_U, for a consumer it is the surplus which it gains from a purchase. More precisely, a producer calculates the expected utility of p_{tmp} and os_U by multiplying for each product interval $t_{pr}^{(i)}$ the provided amount of electrical energy $e_U^{(i)}$ by the predicted price $c_{p_{tmp}}^{(i)}$, subtracting the production cost $e_U^{(i)} \cdot c_U^{(i)}$, and summing up the results over all product intervals $t_{pr}^{(i)} \in T_{pr}^{(p_{tmp})}$. Contrary, a consumer determines the expected utility by multiplying for each product interval $t_{pr}^{(i)}$ the amount of demanded electrical energy $e_U^{(i)}$ by the cost $c_U^{(i)}$ which it is willing to pay, subtracting the predicted energy price $e_U^{(i)} \cdot c_{p_{tmp}}^{(i)}$, and summing up the results over all intervals $t_{pr}^{(i)} \in T_{pr}^{(p_{tmp})}$. Because of assumption A_3 specified at the beginning of Chapter 4 stating that all electricity amounts of a product are either positive or negative, the utility of a storage can simply be calculated by means of one of the above formulas according to the unit's behavior in the prescribed product horizon (i.e.

depending on the fact if it is discharging or charging and thus acting as producer or consumer).
Equation 4.1 is consistent with the distribution model applied in the last activity of DYCE in the sense that the latter guarantees producers to receive at least their declared cost $c_U^{(i)}$ and consumers to pay at most $c_U^{(i)}$. This in turn means that the specification of the schedule cost generally depends on the individual preferences of an agent as well as its trading strategy. For instance, in the context of an exchange producers are typically assumed to declare the marginal cost of their units as price of a product. However, the design of an optimal trading strategy is generally highly complex and requires concepts from domains like microeconomics, game theory, or artificial intelligence (for work in this field see for instance [Pow] and related publications). Thus, we leave this topic to future work as discussed in Section 6.2.
Given Equation 4.1, the utility of a template portfolio TP and an operation schedule os_U is defined as

$$utility(TP, os_U) = \sum_{p_{tmp} \in TP} utility(p_{tmp}, os_U), \tag{4.2}$$

where U can be of type producer, consumer, or storage. The above formula enables an agent to identify an optimal pair of template portfolio and operation schedule. Assuming that template portfolio TP comprises n product templates, the corresponding optimization problem is given by

$$\text{maximize} \quad utility(TP, os_U) \tag{4.3}$$

$$\begin{aligned}
\text{subject to} \quad & TP \in \mathcal{P}(P_{tmp}), && (4.4)\\
& os_U \in OSS_U, && (4.5)\\
& \bigcup T_{pr}^{(p_{tmp,i})} \setminus T_{pl} = \emptyset \wedge \bigcap T_{pr}^{(p_{tmp,i})} = \emptyset, 1 \leq i \leq n, && (4.6)\\
& sat_{hc_U}(os_U) = 1 \; \forall \; hc_U \in HC_U. && (4.7)
\end{aligned}$$

By solving the above problem, inflexible agents are able to identify a utility maximizing template portfolio TP for the given fixed operation schedule os_U (as in this case it always holds that $|OSS_U| = 1$), while flexible agents are able to determine a utility maximizing pair of template portfolio TP and operation schedule os_U. Constraints 4.6 and 4.7 ensure that TP satisfies the property of consistency and that os_U is a feasible schedule. Based on the resulting template portfolio, agents are finally able to create an optimal product portfolio PP including the target products p_{tar} for which coalitions to form.
With regard to Equation 4.1, the benefit which a consumer associates with the operation of its unit U, as reflected by schedule cost $c_U^{(i)}$, might be difficult to determine because it depends on the individual preferences of the user. If these are too costly to asses or

simply irrelevant for the given use case, Equation 4.1 and 4.2 can also be transferred into cost functions by setting $c_U^{(i)}$ to 0, i.e.

$$cost(p_{tmp}, os_U) = \sum_{i=0}^{j_{max}} e_U^{(i)} \cdot c_{p_{tmp}}^{(i)}, \tag{4.8}$$

$$cost(TP, os_U) = \sum_{p_{tmp} \in TP} cost(p_{tmp}, os_U). \tag{4.9}$$

I.e., for a product interval $t_{pr}^{(i)}$, the cost resulting from the purchase of a target product being specified on the basis of a given product template p_{tmp} and operation schedule os_U is simply calculated by multiplying the demanded electricity amount $e_U^{(i)}$ by the predicted price $c_{p_{tmp}}^{(i)}$. Again assuming that template portfolio TP comprises n product templates, the corresponding optimization problem allowing consumers to minimize their cost is given by

$$\text{minimize} \quad cost(TP, os_U) \tag{4.10}$$

$$\text{subject to} \quad TP \in \mathcal{P}(P_{tmp}), \tag{4.11}$$

$$os_U \in OSS_U, \tag{4.12}$$

$$\bigcup T_{pr}^{(p_{tmp,i})} \setminus T_{pl} = \emptyset \wedge \bigcap T_{pr}^{(p_{tmp,i})} = \emptyset, 1 \leq i \leq n, \tag{4.13}$$

$$sat_{hc_U}(os_U) = 1 \; \forall \; hc_U \in HC_U. \tag{4.14}$$

Having specified the optimization problem which is related to the activity of product portfolio management, we now consider the associated computational complexity in order to be able to develop an appropriate technique for its solution in the following section. In order to facilitate discussion, we first analyze the task of generating a product portfolio PP for a single operation schedule os_U. The identification of a utility maximizing template portfolio TP which can be used as basis for the specification of PP is a combinatorial problem in which the product templates p_{tmp} of a given template catalog P_{tmp} have to be combined into a valid solution (i.e. a consistent template portfolio TP). Without the consideration of any constraints, a catalog of n templates generally results in a search space of 2^n valid and invalid template combinations. Because of the exponential growth, for bigger catalog sizes the problem is thus impossible to solve through an exhaustive search for the optimal solution. For instance, with regard to the exemplary catalog of the EPEX SPOT as shown in Figure 4.2, the number of 41 product templates results in a search space of $2^{41} = 2\,199\,023\,255\,552$ template combinations. As this complexity only refers to a single operation schedule os_U, it even rises for flexible units which operation schedule spaces OSS_U comprise large numbers of feasible schedules. In such cases, the identification of an optimal solution through exhaustive search involves the determination of a template portfolio TP for each schedule $os_U \in OSS_U$. It follows that a solution of the above optimization problem requires an appropriate heuristic which allows to efficiently approximate a theoretical optimum.

4.1.2 Activity Description

Figure 4.3 shows the activity of product portfolio management and its relation to the other three DYCE activities[3] including a specification of the corresponding guard conditions[4]. Because for typical sizes of template catalogs it is computationally impossible to generate a utility maximizing product portfolio, the process integrates an algorithm which efficiently approximates an optimal solution. In the following, we refer to this heuristic as COPE[5] – Combinatorial Optimization for Portfolio Enhancement. Its input is generated by an agent in the course of the first three actions of the activity, where in the first step a template catalog P_{tmp} is created which is used as basis for the identification of a utility optimizing template portfolio TP. This task is particularly influenced by the type of market the agent participates in and the related rules for product specification. If the latter prescribe a trade of standardized products (like in case of an exchange), a template catalog is typically defined by the market institution itself and may simply be queried from a market agent providing corresponding functionality. However, if the rules allow for a trade of non-standardized products, product templates have to be created by an agent itself which can be derived from the given contract conditions. Assume for instance a bilateral market in which a consumer offers to buy 100 kWh at cost 0.1 €/kWh from 03:00 pm to 04:00 pm on a considered day. A producer can then add a corresponding product template $p_{tmp} = \{hc_{p,1}, hc_{p,2}, hc_{p,3}, hc_{p,4}\}$ to its template catalog, where

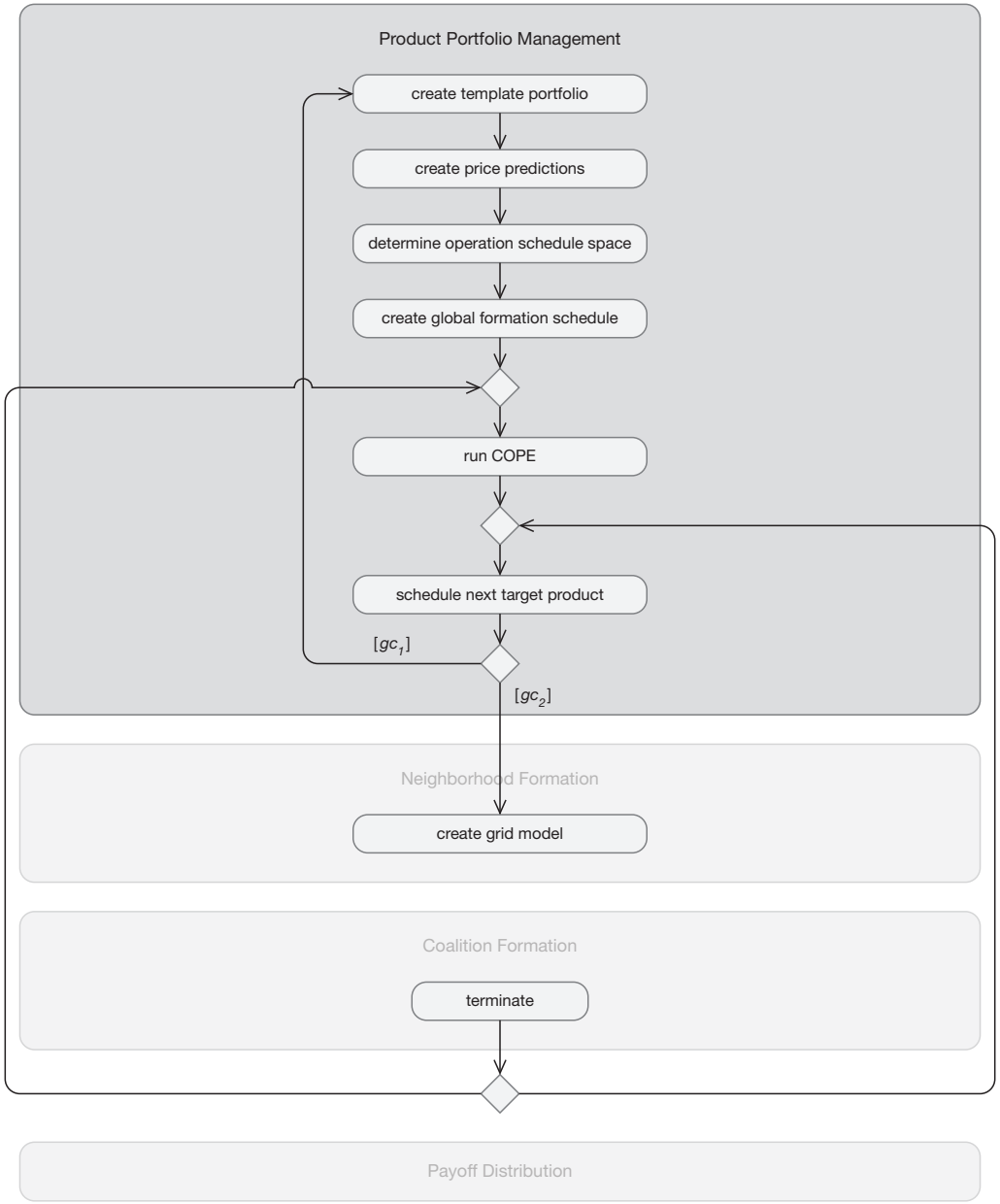
$$
\begin{aligned}
hc_{p,1} &: T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq 95\},\\
hc_{p,2} &: T_{pr}^{(p)} = \{t_{pr}^{(60)}, \ldots, t_{pr}^{(63)}\},\\
hc_{p,3} &: \forall t_{pr}^{(i)} \in T_{pr}^{(p)} : e_p^{(i)} = 100\,\text{kWh},\\
hc_{p,4} &: \forall t_{pr}^{(i)} \in T_{pr}^{(p)} : c_p^{(i)} = 0.1\,€/\text{kWh}.
\end{aligned}
$$

In comparison to the example of a peak load template given in Section 3.3, the above template restricts the electricity amount and cost to the predefined values. Using this approach, agents can flexibly integrate various markets of different type into their decision making process by adding appropriately specified templates to catalog P_{tmp}. The only requirements which have to be fulfilled with regard to the COPE algorithm are first that all added templates have to comprise a product constraint prescribing a specific product horizon (which obviously has to hold for any template) and second that all templates have to possess a unique descriptor making them identifiable throughout the optimization process even if their product horizons are equal.

[3] For the sake of comprehensibility, the following activity diagrams specify the currently considered DYCE activity in full detail while restricting the description of the other ones to those actions which are related to the considered activity. Moreover, guard conditions are only defined for all inner and outgoing control flows of the considered activity.

[4] In the following, most guard conditions are specified by means of boolean methods reflecting conditions which are described in more detail in the text. If newly introduced, these are defined at the end of the tables listing the guard conditions.

[5] pronounced as [koʊp]



ID	condition
gc_1	$p_{tar} \neq null \wedge t_{f,p_{tmp}} = t_{now}$
gc_2	$p_{tar} = null$

Figure 4.3: Activity of product portfolio management including guard conditions.

As second action of the activity, an agent creates price predictions $c^{(i)}_{p_{tmp}}$ for all product templates p_{tmp} of the previously specified catalog. As described in the previous section, the forecasts are used in the course of the optimization process in order to determine the utility of a given pair of template portfolio TP and operation schedule os_U (cf. Equation 4.2). While in case of offers with defined prices these can be directly derived from the template constraints (such as in the example above), predictions for yet undefined prices require appropriate forecasting techniques like statistical or simulation-based approaches. For instance, with regard to the exemplary peak load template given in Section 3.3 which generally allows market participants to specify prices in the range from -500 €/MWh to 3000 €/MWh, an agent would be required to make a prediction of the final clearing price. As forecasting techniques are an own topic of research and thus exceed the scope of this thesis, we refer to related literature for a comprehensive coverage [Wer14, Nii06, HTWI09].
In the course of the third action of product portfolio management, an agent creates the last input of the COPE algorithm in the form of the operation schedule space OSS_U of its controlled unit U. To this end, it predicts the operational behavior of the unit by determining all schedules which are feasible within the considered planning horizon. Depending on the type of unit, these forecasts have to account for controllable factors like scheduled events in a production process or uncontrollable factors like future weather conditions. As agents may deviate from the predictions after coalitions have formed and contracts were concluded, appropriate techniques have to be applied in order to ensure a physical fulfillment according to the stipulated conditions. Specific models for predicting the operational behavior of different unit types can be found in related literature (see for instance [CCN+08, LSH+11]).
Before executing the COPE algorithm, an agent first performs the fourth action of product portfolio management and creates a formation schedule in order to organize the formation processes which are associated with the product templates of its created template portfolio. As described in Section 3.3, all formation activities are generally conducted within formation period t_{CF} which precedes the currently considered planning horizon T_{pl}. Because different agents potentially specify different target products for a given product template, formation processes are generally scheduled with regard to the templates, i.e. for each product template p_{tmp} the formation schedule finally specifies a formation interval $t_{f,p_{tmp}}$ in which coalition formation for corresponding products takes place. In this context, Figure 4.4 depicts two general approaches for the temporal organization of the formation intervals using the catalog of the EPEX SPOT as an example. As shown, these can either be arranged in sequential or parallel order meaning that the corresponding formation activities are either conducted one after the other or simultaneously throughout formation period t_{CF}. With regard to DYCE, we generally choose the former approach as a sequential scheduling provides the advantage that the coordination and communication efforts are distributed across the whole formation period. As agents potentially specify a high number of different target products, a parallel execution of all formation processes could soon become inefficient. As further advantage, a sequential scheduling allows agents to redefine their product portfolio if the formation of a coalition

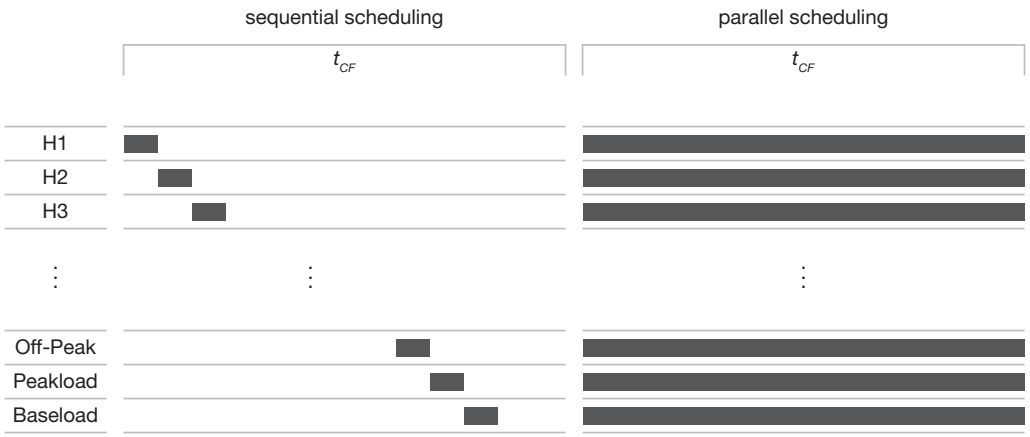


Figure 4.4: Sequential vs. parallel template scheduling.

for a specific target product was unsuccessful. To illustrate, assume for instance that the product templates of the EPEX SPOT are arranged as depicted on the left hand side of Figure 4.4 and an agent was unable to form a coalition for its target product $p_{tar,Off-Peak}$ in the corresponding formation interval $t_{f,p_{Off-Peak}}$. As in this case formation period t_{CF} has not ended yet, the agent is able to generate a new product portfolio by removing product template $p_{tmp,Off-Peak}$ from its template catalog P_{tmp} and executing the COPE algorithm again (a detailed description of this approach is given in Section 4.3.2.4). This kind of redefinition is not possible if the templates are scheduled in parallel as in this case all formation intervals end at the same time.

With regard to the actual creation of the formation schedule, in the context of DYCE we assume that the formation intervals for product templates of mediated markets are prescribed by the institutions themselves just like other trading-relevant dates such as the order book closure of a specific auction. For instance, the EPEX SPOT could arrange the templates of its template catalog as indicated on the left hand side of Figure 4.4. In this case, the formation intervals should not occupy the whole formation period t_{CF} in order to allow agents to schedule further formation processes for bilateral target products. The corresponding intervals can either be negotiated by the agents in a distributed fashion or scheduled by an authorized institution allowing participants to register individual product templates. These approaches finally enable an agent to specify a formation schedule including intervals for all templates of its previously specified template portfolio.

As next action of product portfolio management, an agent then executes the COPE algorithm in order to solve optimization problem 4.3 and identify a product portfolio PP and operation schedule os_U which maximize its expected utility. Generally, the heuristic is independent from the type of a unit and can thus be executed by producers, consumers, and storage likewise. In the following, we provide a detailed description of the algorithm's functionality and illustrate its application by again using our exemplary use case

assuming that the producer intends to create a product portfolio based on the template catalog of the EPEX SPOT.
Algorithm 4.1 shows the pseudocode of the heuristic along with the required input and generated output. In addition to the data created in the course of the first three actions (i.e. template catalog P_{tmp}, price predictions $C_{P_{tmp}}$, and operation schedule space OSS_U), input parameter *drawnOS* specifies the number of operation schedules which are drawn from OSS_U in the course of the optimization process for evaluation. Thus, it generally holds that $drawnOS \leq |OSS_U|$. Furthermore, parameter $P_{tmp,base}$ specifies a consistent base catalog comprising those templates of P_{tmp} which product horizons are as long as the smallest tradable interval. With regard to the template catalog of the EPEX SPOT, the base catalog is for instance given by the twenty-four single hour templates as depicted at the top of Figure 4.2. Note that the templates form a consistent template catalog as the prescribed product horizons cover the whole planning horizon (i.e. all 24 hours) and are temporally disjunct. A base catalog is a general prerequisite for the application of the COPE algorithm but typically exists in every competitive market in which agents have access to an exchange. As described in more detail in Section 4.3.2.4, in the course of the overall process agents replace parts of the base catalog by templates for which target products successful coalitions could be formed. This measure guarantees that the agents are finally committed to the fulfillment of their initial target products only and that these are not replaced by other target products in the course of a redefinition of their product portfolio. The returned output of COPE is given by a product portfolio PP and operation schedule os_U which in conjunction optimize the agent's utility.
The functionality of the heuristic, as implemented by the listed code, can generally be divided into three different parts. While the first part initializes all relevant variables with appropriate values (line 1-4), the second one identifies a utility optimizing template portfolio TP and operation schedule os_U (line 5-26). Based on the template portfolio, the last part then creates a product portfolio PP and returns it along with os_U as output (line 27-33).
More precisely, the heuristic starts by initializing variable $P_{tmp\backslash base}$ with a reduced version of the template catalog which is created by removing all templates of base catalog $P_{tmp,base}$ from template catalog P_{tmp}. Moreover, variable TP and os_U store the utility optimizing template portfolio and operation schedule and are initialized with the base catalog and the value *null*, respectively. The utility which is associated with both is finally stored by variable $util_{os_u}$ which is initially set to 0 (line 1-4). As the following optimization process iteratively searches for better solutions, TP, os_U, and $util_{os_u}$ always store the currently best result at each point in time.
Having finished the initialization of the global variables, the algorithm starts the identification of a utility optimizing pair of template catalog and operation schedule by sorting the product templates of the reduced template catalog according to the following rule: For any two templates $p_{tmp,A}$ and $p_{tmp,B}$ with identifiers id_A and id_B and product horizons

Algorithm 4.1: Combinatorial Optimization for Portfolio Enhancement

input : P_{tmp}, $C_{P_{tmp}}$, OSS_U, $drawnOS$, $P_{tmp,base}$
output: PP, os_U

$P_{tmp\backslash base} \leftarrow P_{tmp} \setminus P_{tmp,base}$
$TP \leftarrow P_{tmp,base}$
$os_U \leftarrow null$
$util_{os_U} \leftarrow 0$
$sort(P_{tmp\backslash base})$
for $i \leftarrow 1$ **to** $drawnOS$ **do**
  $TP_{os'_U} \leftarrow P_{tmp,base}$
  $os'_U \leftarrow drawRandomOS\ (OSS_U)$
  $util_{os'_U} \leftarrow utility\ (TP_{os'_U}, os'_U)$
  for $j \leftarrow 1$ **to** $|P_{tmp\backslash base}|$ **do**
    $p_{tmp} \leftarrow P_{tmp\backslash base}$ [j]
    if $areCompatible(p_{tmp}, os'_U)$ **then**
      $TP'_{os'_U} \leftarrow replace(TP_{os'_U}, p_{tmp})$
      $util'_{os'_U} \leftarrow utility\ (TP'_{os'_U}, os'_U)$
      if $util'_{os'_U} \geq util_{os'_U}$ **then**
        $TP_{os'_U} \leftarrow TP'_{os'_U}$
        $util_{os'_U} \leftarrow util'_{os'_U}$
      end
    end
  end
  if $util_{os'_U} \geq util_{os_U}$ **then**
    $TP \leftarrow TP_{os'_U}$
    $os_U \leftarrow os'_U$
    $util_{os_U} \leftarrow util_{os'_U}$
  end
end
foreach $p_{tmp} \in TP$ **do**
  if $areCompatible(p_{tmp}, os_U)$ **then**
    $p_{tar} \leftarrow tarProduct(p_{tmp})$
    $add\ (PP, p_{tar})$
  end
end
return PP, os_U

$T^{(p)}_{pr,A} = \{t^{(0)}_{pr,A}, \ldots, t^{(j_{max})}_{pr,A}\}$ and $T^{(p)}_{pr,B} = \{t^{(0)}_{pr,B}, \ldots, t^{(k_{max})}_{pr,B}\}$, $p_{tmp,B}$ is put after $p_{tmp,A}$ if one of the following conditions hold:

$$|T^{(p)}_{pr,A}| < |T^{(p)}_{pr,B}|, \tag{4.15}$$

$$|T^{(p)}_{pr,A}| = |T^{(p)}_{pr,B}| \wedge t^{(0)}_{pr,A} < t^{(0)}_{pr,B}, \tag{4.16}$$

$$|T^{(p)}_{pr,A}| = |T^{(p)}_{pr,B}| \wedge t^{(0)}_{pr,A} = t^{(0)}_{pr,B} \wedge id_A < id_B. \tag{4.17}$$

I.e., in line 5 function $sort(P_{tmp\backslash base})$ arranges the templates according to the lengths of the prescribed product horizons (4.15) and, if two lengths are equal, according to the start times of the latter (4.16). In case even these are identical, the templates are sorted according to their unique identifier (4.17). Figure 4.2 already shows the catalog of the EPEX SPOT in sorted order in which the single hour templates are followed by the block templates of longer length. Based on the sorted catalog, the algorithm then starts an iterative optimization process by entering a for loop in which it identifies a utility optimizing pair of template portfolio and operation schedule. I.e., in each of *drawnOS* iterations, it draws a random schedule from the operation schedule space of the given unit and then generates a corresponding optimized template portfolio. If the identified pair yields a utility greater than or equal to the currently best solution, the values are stored as new best result. More precisely, line 7-9 initialize the required variables by setting the newly to be generated template portfolio $TP_{os'_U}$ to the base catalog, drawing a random schedule os'_U from schedule space OSS_U via function $drawRandomOS(OSS_U)$, and calculating the associated utility $util_{os'_U}$ via function $utility(TP_{os'_U}, os'_U)$ (cf. Equation 4.2). Within the following for loop, the algorithm then identifies an optimized template portfolio $TP_{os'_U}$ for schedule os'_U. To this end, it iterates over the elements of the sorted reduced template catalog and replaces temporally overlapping elements in $TP_{os'_U}$ if the former provide a utility greater than or equal to the latter. I.e., the for loop starts at the first position of $P_{tmp\backslash base}$ and sets the corresponding element as currently considered template p_{tmp} (line 11). Next, function $areCompatible(p_{tmp}, os'_U)$ evaluates if the template is compatible with schedule os'_U in terms of assumption A_3. I.e., it verifies if within the product horizon which is prescribed by the template all of the schedule's electricity amounts $e^{(i)}_U$ are either strict positive or strict negative. If true, function $replace(TP_{os'_U}, p_{tmp})$ creates a new portfolio $TP'_{os'_U}$ by first replacing those templates of $TP_{os'_U}$ with p_{tmp} which are temporally covered by the very same and then filling up resulting gaps with templates from the base catalog. To illustrate, Figure 4.5 shows the initial and final state of an exemplary replacement. The upper part of the figure depicts those templates of portfolio $TP_{os'_U}$ which are to be exchanged by the currently considered template $p_{tmp,Morning}$. Because the product horizon of $p_{tmp,EarlyMorning}$ also covers hour 5 and 6, the resulting gaps are filled up with the corresponding templates from the base catalog (i.e. $p_{tmp,H5}$ and $p_{tmp,H6}$) as shown in lower part of the figure.
After replacement, line 14 determines the utility $util'_{os'_U}$ of portfolio $TP'_{os'_U}$ and schedule os'_U. If this is greater than or equal to utility $util_{os'_U}$ which is associated with the former portfolio $TP_{os'_U}$, the corresponding variables are updated to the new values (line 15-18). An update in case of equality means that longer product horizons are generally

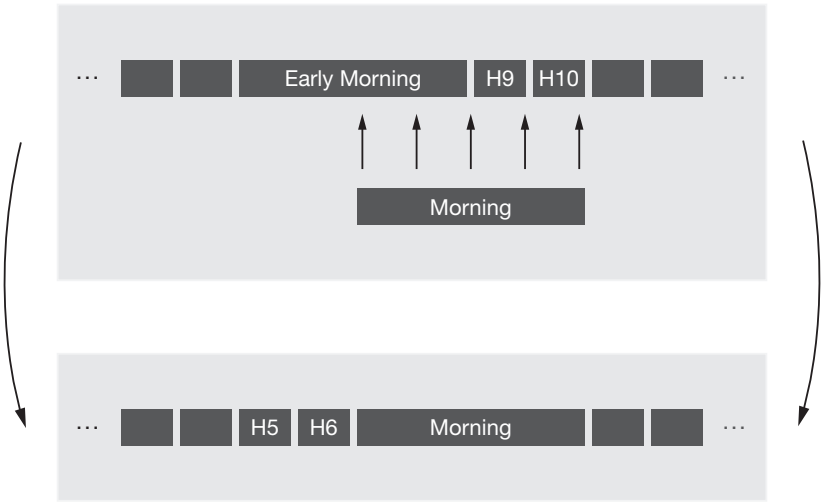


Figure 4.5: Creating portfolio $TP'_{os'_U}$ from $TP_{os'_U}$ by template replacement.

preferred over shorter ones if the related utilities are the same. The reason for this design decision is due to fact that a constant operation of a unit is typically more economical. The algorithm continues by iterating over the reduced template catalog until the last element has been evaluated in which case the best possible template portfolio $TP_{os'_U}$ for schedule os'_U has been identified. Afterwards, it checks if the associated utility $util_{os'_U}$ is greater than or equal to the utility of the currently best solution and, if true, updates all variables by setting them to the new results (line 21-25). An update in case of equality ensures that in the first iteration the global variables are set to a valid solution. The algorithm proceeds with the optimization process until *drawnOS* schedules have been evaluated and a utility optimizing pair of template portfolio and operation schedule has been identified.

The resulting template portfolio is finally used for the specification of a product portfolio PP (line 27-32). To this end, the algorithm evaluates for each template $p_{tmp} \in P_{tmp}$ if it is compatible with the identified schedule os_U. This test is again required as the final portfolio may include templates from the base catalog which were not considered by the if statement in line 12. If a template p_{tmp} is compatible, function $tarProduct(p_{tmp})$ creates a target product p_{tar} which satisfies all hard constraints of p_{tmp} (line 29). To this end, it determines the target values of the product (i.e. electricity amount $e^{(i)}_{p_{tar}}$, error $err^{(i)}_{p_{tar}}$, and cost $c^{(i)}_{p_{tar}}$) based on the trading strategy of the agent. Generally, the specified values have an impact on both the search for like-minded cooperation partners in the course of the following formation process and the success of a trade. For instance, specifying a high target cost increases the chance of a producer to obtain a higher revenue, but decreases the probability to find other participants striving for the same target value and a trading partner willing to pay the price. However, as already mentioned, the design of a trading

strategy extends the scope of this thesis why we leave the assessment of optimal target values to future work. After its specification, p_{tar} is added to the product portfolio using function $add(PP, p_{tar})$. Having processed all templates of the template portfolio, the algorithm finally terminates by returning product portfolio PP and operation schedule os_U as output.

In order to analyze its performance, we next determine the time complexity of COPE in terms of the O-notation. The input data which are relevant for the discussion are given by those parameters which variability has an impact on the algorithm's running time. These are given by template catalog P_{tmp} and the number of drawn schedules $drawnOS$, whereas the price predictions $C_{P_{tmp}}$ and the operation schedule space OSS_U can be considered constant as their variability is practically restricted by P_{tmp} and $drawnOS$, respectively. In this regard, two special cases arise if either the trading activities of an agent are limited to a fixed set of markets with predefined template catalogs or the schedule space of its controlled unit is fixed. In these situations, the corresponding parameters are constant as in the former case the template catalog never changes and in the latter case the number of drawn schedules is restricted because of the inflexibility of the unit. Thus, the most complex case is given if COPE is applied by an agent which constantly specifies varying catalogs (e.g. when participating in a bilateral market) and controls a flexible unit which schedule space comprises varying numbers of feasible schedules over time.

In order to determine the time complexity of COPE for this case, it is required to identify those segments of the algorithm's code which depend on the variable parameters. With regard to the following discussion, let $m = |P_{tmp}|$ and $n = drawnOS$. While all statements in line 1-4 take constant time, the sorting of the reduced template catalog in line 5 can be accomplished by means of a merge sort requiring $O(m \cdot \log m)$ time. Line 6-26 then execute two nested for loops, the outer one iterating over the number of drawn schedules and the inner one over the templates of the reduced template catalog. As all other statements take constant time, the running time of this segment is $O(n \cdot m)$. All remaining code from line 27-33 is again independent from the variable parameters and thus irrelevant for the discussion. All in all, the general worst case time complexity of COPE is thus $O(m + m \cdot \log m + n \cdot m) = O(m \cdot \log m + n \cdot m)$. In this regard, it is worth noting that the size of the template catalog is typically very small compared to the number of drawn schedules, meaning that the running time is particularly determined by parameter n. Moreover, with regard to the above discussed special cases, the time complexity reduces to $O(n)$ if template catalog P_{tmp} is constant and to $O(m + m \cdot \log m) = O(m \cdot \log m)$ if the controlled unit is inflexible. The heuristic thus provides an efficient means for identifying a utility optimizing pair of product portfolio and operation schedule. Further qualitative aspects of the algorithm are examined in the context of evaluation in Section 5.2.

As last action of product portfolio management, an agent finally schedules the next of its target products according to the previously created formation schedule and starts the activity of neighborhood formation when the corresponding formation period begins (guard condition gc_1). However, if there is no more target product left to schedule (for

instance because all formation processes have been successfully finished), the agent updates its planning horizon to the following period (e.g. the next day) and starts a new iteration of the overall process (gc_2).

4.2 Neighborhood Formation

Having scheduled a target product as last action of product portfolio management, an agent starts the activity of neighborhood formation in order to restrict the number of potential cooperation partners to an efficiently manageable size if it initiates negotiations in the following coalition formation. To this end, it applies a distance measure which allows it to identify its nearest neighbors based on characteristics of the underlying power grid. In the following, we cover all aspects related to the activity by first providing a more detailed description of the considered problem along with an analysis of its associated computational complexity. Afterwards, we describe the corresponding solution concepts which allow for an assessment of grid-related distances and the formation of neighborhoods of nearest neighbors.

4.2.1 Problem Specification and Computational Complexity

As second activity of the overall process, an agent a_i performs the task of neighborhood formation in order to limit the set of interaction partners if it initiates negotiations in the following coalition formation. It starts the process for a scheduled target product p_{tar} by creating an internal model of the given power grid G including all connected units U_j and assigned agents a_j. As the resulting data structure also comprises the distance weights $dw(v_{G,k}, v_{G,l})$ which are related to the power lines $\{v_{G,k}, v_{G,l}\}$ of the grid (cf. Definition 3.1), it is used by the agent in order to calculate the physical distances $dis((v_{G,i}, v_{G,j}))$ to all other agents a_j. In the context of DYCE, the distance between two grid nodes $v_{G,i}$ and $v_{G,j}$ is generally defined in terms of the impedances of the connecting power lines. As measure of the opposition which is faced by an electric current flowing through the lines, these provide a reasonable distance criterion c_{dis} for determining the corresponding weights $dw(v_{G,k}, v_{G,l})$[6]. As the created grid model also provides information about all other participants a_j in the market, the agent uses the knowledge in order to assess the corresponding trust values $trust((t, a_i, a_j))$ in a distributed fashion allowing it to evaluate the trustworthiness of potential cooperation partners in the course of coalition formation. Furthermore, the agent chooses its initial role which it assumes in the upcoming negotiations by determining if it either proactively initiates and responds to formation requests or responds only. In the former case, it builds a neighborhood N by identifying the s_N nearest neighbors according to the previously calculated distances and adding them as potential cooperation partners to N.

[6] Note that according to Definition 3.9, distance weights can generally be calculated based on multiple distance criteria. I.e., while in this thesis we only consider the impedances of the lines, further aspects can be taken into account if required by the given use case.

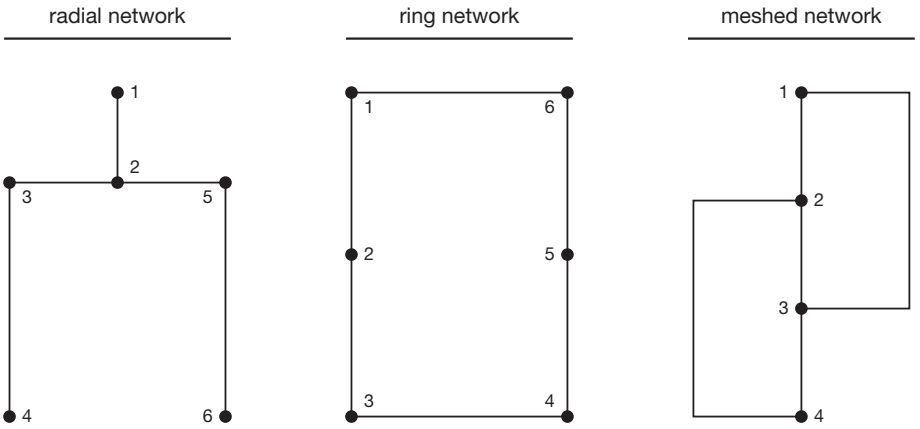


Figure 4.6: Graphs representing a radial, ring, and meshed network.

With regard to the activity's computational complexity, the most challenging task of neighborhood formation is given by the assessment of the distances to the other agents in the market. As already indicated above, an agent a_i determines the distance to another agent a_n based on the complex impedances of the power lines that link the grid nodes to which their units are connected. I.e., given that $gn(u(a_i)) = v_{G,i}$ and $gn(u(a_n)) = v_{G,n}$, for distance assessment it has to consider the topology of the grid and in particular the power lines $\{v_{G,i}, v_{G,j}\}, \ldots, \{v_{G,m}, v_{G,n}\}$ making up the paths between $v_{G,i}$ and $v_{G,n}$. To illustrate, Figure 4.6 shows minimal graphs of the three grid topologies which are applied in the context of power provision, i.e. a radial network, a ring network, and a meshed network. As depicted, in case of a radial network there is only a single path between any two grid nodes $v_{G,s}$ and $v_{G,t}$, whereas in case of a ring network there are always two parallel paths. In a meshed network, the number depends on the considered start and end node which might be connected by one or more paths. Now suppose we want to assess the distance between node $v_{G,1}$ and $v_{G,4}$ for each of the given examples. The complexity which is associated with this task then differs according to the given topology. More precisely, as in case of the radial network all power lines are connected in series, we can simply add their complex impedances $\underline{Z_i}$ which sum up to the total impedance $\underline{Z_{ser}}$. I.e., for a path with n power lines $\{v_{G,k}, v_{G,l}\}, \ldots, \{v_{G,o}, v_{G,p}\}$, the latter is calculated as [KSW13]

$$\underline{Z_{ser}} = \sum_{i=1}^{n} \underline{Z_i}. \tag{4.18}$$

However, for the ring network the above formula does not hold because $v_{G,4}$ can be reached from $v_{G,1}$ by more than one path and the lines are connected in parallel. In this case, the total impedance $\underline{Z_{par}}$ can be calculated by determining the impedances of all

paths and adding their reciprocals which sum up to the reciprocal of $\underline{Z_{par}}$. I.e., for $m > 2$ parallel paths, the total impedance is calculated as [KSW13]:

$$\underline{Z_{par}} = \left(\sum_{i=1}^{m} \frac{1}{\underline{Z_{ser,i}}} \right)^{-1} = \left(\sum_{i=1}^{m} \frac{1}{\sum_{j=1}^{n} \underline{Z_j}} \right)^{-1}. \tag{4.19}$$

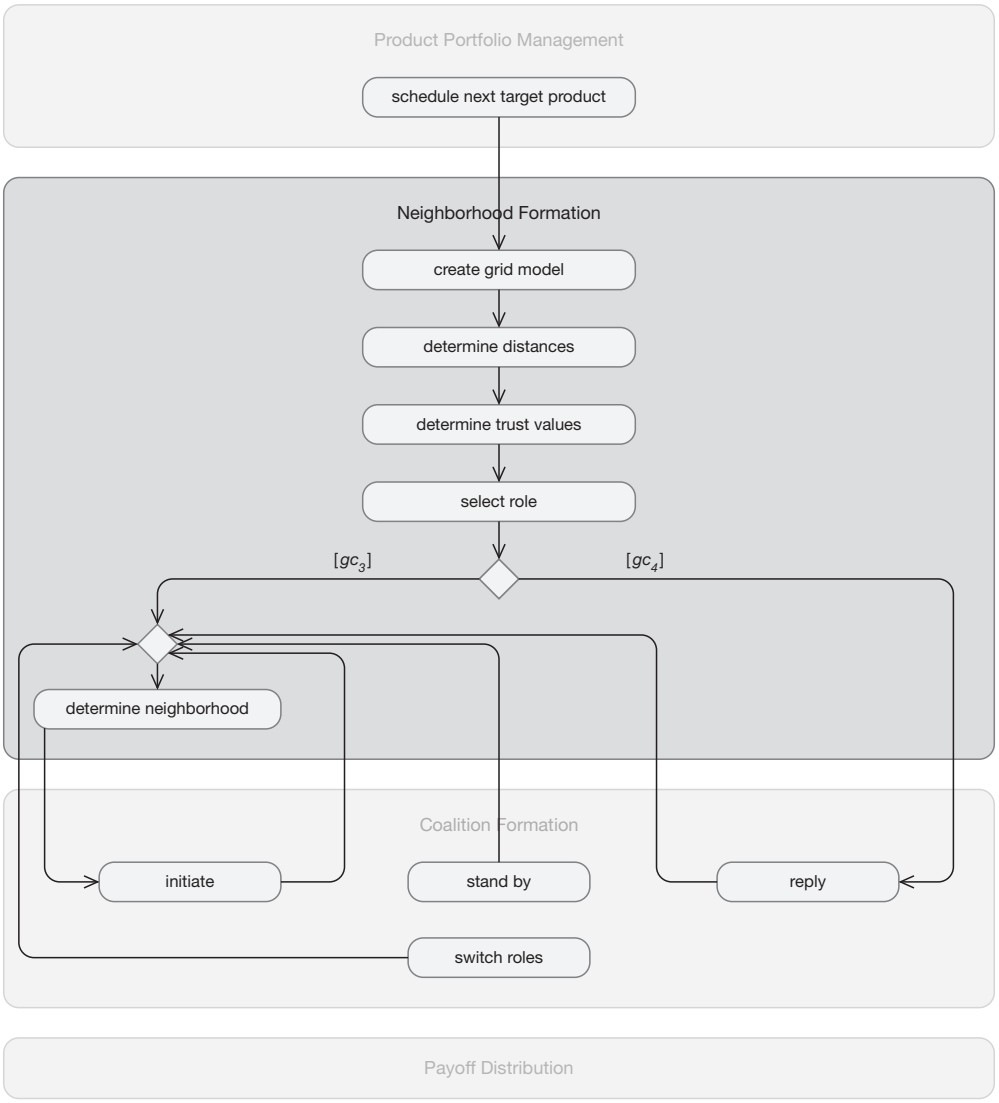
As in radial and ring networks the number of paths between two grid nodes is inherently restricted because of the corresponding topological structures, distances can be calculated with a low computational effort even in case of large power grids. However, because in meshed networks power lines are typically not solely connected in series or in parallel, the above formulas generally do not hold and an alternative method for the calculation of impedances has to be used. Such an approach requires to make assumptions with regard to the applied voltage as well as the produced and consumed electrical energy wich is fed into and taken from the grid. However, as DYCE is designed as a distributed method, this kind of global information is not available to the agents. Moreover, in meshed networks the number of paths between two distant grid nodes is typically very high which makes the calculation of impedances a highly complex task. These reasons motivate the development of an alternative technique which allows agents to determine the distances to other participants in an efficient way as described in more detail next.

4.2.2 Activity Description

The activity of neighborhood formation and its relation to the other DYCE activities is depicted by the UML diagram in Figure 4.7. As first action of the process, an agent initially creates a model of the power grid G which represents the place of delivery of the market M in which it intends to trade its pursued target product p_{tar}. The information which is required for this task can either be retrieved from grid agents which are operated by the responsible system operators or transferred offline by legitimate service providers. The data have to comprise on the one hand information about the grid topology including the complex impedances $\underline{Z} \in \mathbb{C}$ of the power lines and on the other hand information about the connected units U and their assigned agents a. After retrieval, the agent creates the grid model by mapping the data to a weighted graph in the sense of Definition 3.1, where the units and agents are linked to the grid nodes through a grid node assignment gn and unit assignment u, respectively. The weights of the graph are determined by assessing the power lines' apparent impedances $Z \in \mathbb{R}$ which can be derived from the complex impedances $\underline{Z} = R + jX$ through calculation of their absolute values [KSW13]:

$$Z = |\underline{Z}| = \sqrt{R^2 + X^2}. \tag{4.20}$$

The reason for using the apparent instead of the complex impedances is due the approach which is used for distance assessment later in the process requiring the weights to be comparable. After creation, the model is finally stored in local memory and updated before each new formation process.



ID	condition[a]
gc_3	*initiator*()
gc_4	¬*initiator*()

[a] *initiator*() returns true if the agent is initiator, false otherwise.

Figure 4.7: Activity of neighborhood formation including guard conditions.

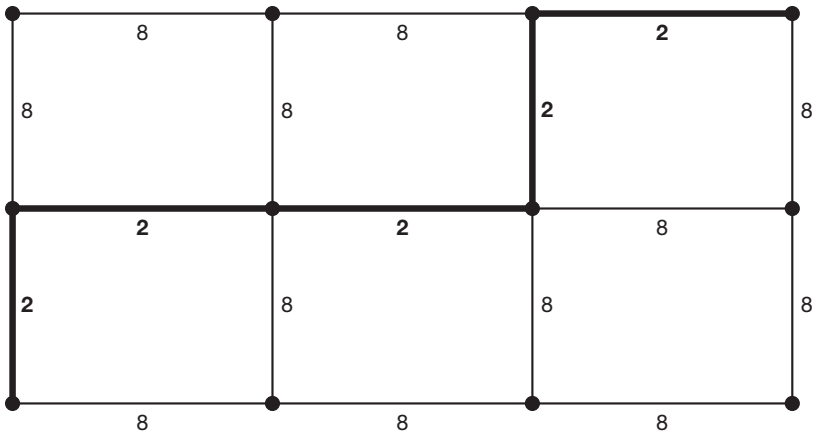


Figure 4.8: Shortest path between two grid nodes.

A special situation arises if the target product is to be provided in a specific grid section which might be for instance the case if the agent participates in a market being designed for the trade of localized products that are used by system operators for congestion management. Then, the grid model (and thus the set of potential cooperation partners) is restricted to the corresponding region. Similarly, if the power grid covers an unreasonably large area, the data structure can be limited to a selected area including the units which are considered relevant for the following formation process.
As second action of neighborhood formation, an agent determines the distances to all other participants in the market based on the generated grid model. As motivated in the previous section, this task requires an appropriate assessment technique allowing agents to perform the calculations efficiently and independent from the given grid topology. Thus, we apply concepts from graph theory in order to approximate the impedances between grid nodes based on the previously created grid model. To illustrate, Figure 4.8 shows a weighted graph reflecting the topology of an exemplary meshed network. Based on this kind of data structure, an agent determines the distances to other participants by identifying the shortest paths to the corresponding grid nodes, i.e. it identifies those connections which weights yield the lowest sum of their lines' apparent impedances as indicated by the bold line. Given a graph (V_G, E_G), the distance between two grid nodes $v_{G,i}$ and $v_{G,j}$ is thus defined as function $dis : V_G \times V_G \rightarrow \mathbb{R}_{\geq 0}$, where

$$dis((v_{G,i}, v_{G,j})) = \begin{cases} \sum_{\{v_{G,m}, v_{G,n}\} \in sp_{ij}} dw(v_{G,m}, v_{G,n}) & \text{if } i \neq j, \\ 0 & \text{else,} \end{cases} \tag{4.21}$$

and sp_{ij} is the shortest path between $v_{G,i}$ and $v_{G,j}$. Identifying the distances to all other participants can be accomplished by applying Dijkstra's shortest path algorithm running in $O(|E| + |V| \cdot \log |V|))$ time [CLRS09]. The resulting values are used by an agent in

order to create a sorted list of the participants which allows it to assess and extend its neighborhood later in the process.
As an alternative approach, the previous two actions could also be performed by a central yellow pages agent providing the information to all market participants upon request. However, although such a service would reduce the computational cost in favor of the agents and supersede the need to transmit potentially confidential grid data, it introduces a single point of failure which avoidance is a main reason for applying distributed systems in the first place. Therefore, DYCE assigns the tasks to the agents, particularly against the background that changes in a grid typically occur very seldom and the grid model has thus to be updated rarely. Nevertheless, if regulations prohibit a transfer of grid-related information, a yellow pages service would be a reasonable solution to the problem.
In the course of the third action of neighborhood formation, an agent determines the trust values of all other participants allowing it to assess the trustworthiness of potential cooperation partners in the course of coalition formation. Generally, trust constitutes a research topic on its own which has been widely investigated in different scientific fields throughout the last decades [RSBC98]. In related work, it is often understood as a multi-dimensional concept comprising different facets like reliability or honesty which contribute to the trustworthiness of an entity. However, which criteria are actually considered relevant often varies according to the context in which the topic is discussed. For instance, in [SKL+10], Steghöfer et al. identify the six facets functional correctness, safety, security, reliability, credibility, and usability as relevant aspects for organic computing systems. In contrast to reputation, which represents a group's collective view of another party's trustworthiness [JIB07], trust can be considered as a subjective belief of a single entity. This belief evolves over time according to the experiences which the entity gains in the course of cooperations with the trusted party. As trust is a general precondition for any interaction between two cooperation parters, it is a crucial topic in the context of multiagent systems as well [RHJ04].
With regard to the problem of coalition formation in electricity markets, criteria which contribute to the trustworthiness of an agent are for instance given by the security measures which it applies for the protection of its underlying ICT infrastructure or the degree of reliability which its unit provides in the course of product fulfillment. However, as the design of a comprehensive trust model extends the scope of this thesis and is currently topic of other research [RUS14], in the following we use the abstract model specified in Definition 3.8. In order to determine its trust in another agent a_j at a specific point in time t in the form of a trust value $trust((t, a_i, a_j))$, an agent a_i can generally take two different approaches according to its prior experiences with a_j. First, if both agents have already cooperated in the course of previous formation activities, a_i can use the information and determine the value based on the outcomes. In contrast, if no interactions have taken place yet and related knowledge is not available to the agent, it can acquire new information for the assessment by querying other participants with regard to their trust in a_j. This approach is based on the transitive trust principle [JP05] which is depicted in more detail in Figure 4.9. In the example, there is first only a direct trust relationship

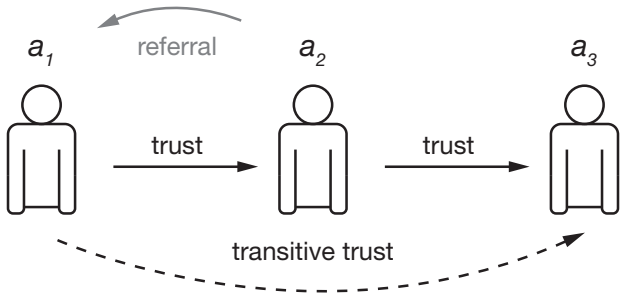


Figure 4.9: Transitive trust principle (adopted from [JP05]).

between agent a_1 and a_2 as well as agent a_2 and a_3. The basic idea of the transitive trust principle is now that agent a_1 can establish an indirect trust relationship to a_3 by asking a_2 about his trust in a_3 and combining the answer with its own trust in a_2. Through the exchange of related information, participants are thus able to assess trust values for (in this sense) unknown agents in a distributed fashion. However, in order to generate semantically consistent results, all participants have to apply the same trust model which uses an identical set of trust criteria c_{trust} and the same formula for calculating the trust values $trust((t, a_i, a_j))$. This requirement can be enforced on global system level by defining an appropriate regulatory hard constraint hc_a.
Having determined the trust values of all other participants, an agent executes the next action of neighborhood formation and chooses its initial role which it assumes in the course of the upcoming interactions for coalition formation. More precisely, it decides if it starts the activity as an initiator which sends and replies to formation requests or takes on the role of a responder which responds to requests only (a more detailed description of both roles is given in Section 4.3.2.3). Generally, the ratio of initiators to the whole number of agents is an important system parameter which is formally defined as

$$r_{init} = \frac{|A_{init}|}{|A|}, \tag{4.22}$$

with $A_{init} \subseteq A$ being the set of initiating agents. As this ratio has an impact on criteria like the global communication cost or runtime[7], it has to be chosen with care and specified globally by means of a regulatory hard constraint hc_a. Given a specific ratio, an agent can determine its initial role by assuming the role of an initiator with a probability of $P(A) = r_{init}$ (or, analogously, taking on the role of a responder with a probability of $P(\bar{A}) = 1 - r_{init}$). For instance, given a prescribed global ratio of $r_{init} = 0.2$, an agent assumes the role of an initiator with a probability of 20 %. This way, in different formation processes the task of initiating is not steadily assigned to the same agents which allows on the one hand to distribute associated overhead costs (like increased computations) to all participants and on the other hand to reduce the risk that untrustworthy agents exploit associated responsibilities (like payoff distribution) for their own good.

[7] The effects of parameter r_{init} are investigated in more detail in the context of evaluation in Section 5.3.

If an agent takes on the role of an initiator, it finishes the second DYCE activity by determining its neighborhood N which restricts the set of potential communication partners in the course of the following formation process and thus reduces communication and computational cost (guard condition gc_1). However, if formation attempts with current neighbors fail, a neighborhood can still be expanded by a specified extension count ext_N. Initially, its size s_N is thus set to ext_N, where an agent identifies the corresponding set of nearest neighbors using its previously created list of market participants. As this stores the participants in sorted order, it generally allows for an efficient creation and extension of a neighborhood. In contrast, if an agent assumes the role of a responder, it directly starts the next activity (gc_2) and determines its neighborhood later in the process if it changes roles and becomes an initiator.

4.3 Coalition Formation

Based on its initial interaction role selected in the course of neighborhood formation, an agent starts the third DYCE activity throughout which it forms coalitions with other participants in order to fulfill its scheduled target product. In what follows, we provide a detailed description of the process by first specifying the considered problem along with the associated computational complexity. Afterwards, we present solution concepts which allow for a dynamic evaluation and formation of product-related coalitions. In particular, we describe an interaction protocol which is defined on the basis of well-established standards and specify a process flow enabling agents to form coalitions in a self-organized and fully distributed fashion.

4.3.1 Problem Specification and Computational Complexity

In the course of the third DYCE activity, an agent strives for the formation of a coalition C in order to eventually fulfill its currently scheduled target product p_{tar}. Because this is the goal of all participants in the market, the related interactions result in successively evolving coalitions which are evaluated according to their ability to fulfill the target products pursued by their members. As general optimization goal, the agents strive for a maximization of mean coalition value $\overline{v(C)}$ and hence the global value of coalition structure CS. I.e., the optimization problem which they deal with throughout the formation process is given by

$$\text{maximize } \overline{v(C)} \tag{4.23}$$

$$\text{subject to } C \in \widetilde{CS}, \tag{4.24}$$

$$\forall C, C' \in CS \text{ with } C \neq C' : C \cap C' = \emptyset \wedge \bigcup_{C \in CS} C = A. \tag{4.25}$$

Thus, an optimal solution of the problem is given by a coalition structure CS which coalitions fulfill their target product perfectly. Because of the reasons discussed in Section 3.1, constraint 4.25 ensures that agents only form coalitions which are disjunct with regard to their members.

We discuss the computational complexity of the above optimization problem and show that it is NP-hard by drawing on results from Sandholm et al. who proved in [SLA+99] that finding the optimal coalition structure is NP-complete if the input is given by non-negative coalition values $v(C)$. As the considerations were made in the context of the CSG problem as defined in Equation 1.1, the value of the coalition structure $V(CS)$ was here not defined as the mean but the sum of the coalition values, i.e. $V(CS) = \sum_{C \in CS} v(C)$. The decision problem used for the prove of NP-completeness was given by the question if there is a coalition structure CS which value is equal to or greater than a predefined real number k. This problem is in class NP because a given solution can be verified by calculating $V(CS)$ (as sum of the single coalition values) in polynomial time. The authors proved that under the made assumptions the identification of an optimal coalition structure is NP-complete by reducing the set packing problem to the considered one.
In order to assess the computational complexity of optimization problem 4.23, we can now build upon this proof by showing that first the coalition values in DYCE are always non-negative and second that the verification of a solution to the above decision problem can still be done in polynomial time given our alternative approach for evaluating the coalition structure. Regarding the first aspect, we will see in Section 4.3.2.2 that in the context of our approach the value of a coalition is generally determined by the cumulated contributions of its participating members. Taking into account that agents only start the activity of coalition formation if they are able to contribute to their pursued target products (cf. Section 4.1.2), it follows that they only have a positive impact on coalitions resulting in strict positive values. With regard to the second aspect, it can be easily seen from the equation in Definition 3.17 that the mean coalition value $\overline{v(C)}$ is indeed determinable in polynomial time meaning that the above discussed decision problem still belongs to NP. Given this, it follows that optimization problem 4.23 is NP-complete as well. Thus, because DYCE is intended to be applicable to a large number of agents being in the order of hundreds to thousands, the following section provides an approach for approximating an optimal solution in an efficient and fully distributed way.

4.3.2 Activity Description

The activity of coalition formation is designed as a decentralized process in which agents start as singleton coalitions and successively regroup into optimized coalitions in order to finally fulfill their pursued target products. Before we provide a comprehensive description of the related actions and interactions, we first detail the concept of a contribution in Section 4.3.2.1 and show how multiple contributions are aggregated into the cumulative contribution of a coalition. Afterwards, we present the approach for assessing the coalition value in Section 4.3.2.2 and describe how it promotes the formation of optimized coalitions. Based on the given concepts, we then detail the decentralized formation process by specifying the applied interaction protocol, optimization algorithm and methods for deadlock/livelock prevention in Section 4.3.2.3 and describing the process flow of the activity in Section 4.3.2.4.

4.3.2.1 Contribution and Cumulative Contribution

When forming coalitions, agents aggregate the technical capabilities of their controlled units in order to cooperatively fulfill their target products. According to Definition 3.40, a contribution of an agent a to a product p, with product horizon $T_{pr}^{(p)} = \{t_{pr}^{(0)}, \ldots, t_{pr}^{(j_{max})}\}$, is defined as a function $con_{a,p} : T_{pr}^{(p)} \rightarrow \mathbb{R}^3$ which values $con_{a,p}(t_{pr}^{(i)}) = (e_U^{(i)}, err_U^{(i)}, c_U^{(i)})$ specify the electricity amount $e_U^{(i)}$ that is generated or consumed by the controlled unit U with error $err_U^{(i)}$ at cost $c_U^{(i)}$ in each product interval $t_{pr}^{(i)} \in T_{pr}^{(p)}$. As the operation of a unit is optimized in the course of product portfolio management, the values of the three attributes generally depend on the operation schedule os_U which results as output from the execution of the COPE algorithm. Taking into account the constraint that in each planning interval an agent can only be member of one coalition at a time (cf. Definition 3.11), it follows that $\forall t_{pr}^{(i)} \in T_{pr}^{(p)}$ with $t_{pr}^{(i)} = t_{pl}^{(i)} : con_{a,p}(t_{pr}^{(i)}) = os_U(t_{pl}^{(i)})$, i.e. the entries of a contribution are determined by the corresponding entries of the previously identified schedule.

With regard to the individual attributes, electricity amount $e_U^{(i)}$ specifies the quantity which is generated or consumed by the controlled unit in the corresponding product interval. As this value is only a forecast, error $err_U^{(i)}$ additionally specifies a possible relative deviation from the amount which may occur in the course physical fulfillment due to unforseen events. More precisely, in DYCE we define the error in terms of the root mean square error $RMSE$[8] which is calculated based on the former operational behavior of the unit and put into relation with the provided or consumed electricity amount. I.e., given a data set of n past predictions $e_{U,j}^{(i)}$ and n past measurements $\widetilde{e_{U,j}^{(i)}}$, the error of a contribution in a planning interval $t_{pl}^{(i)}$ is formally defined as

$$err_U^{(i)} = \frac{\sqrt{MSE_U^{(i)}}}{|e_U^{(i)}|} = \frac{\sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(e_{U,j}^{(i)} - \widetilde{e_{U,j}^{(i)}}\right)^2}}{|e_U^{(i)}|}. \tag{4.26}$$

As the above formula squares former deviations and divides the root of their sum by the absolute value of the provided or consumed electricity amount, it generally yields a positive error independent from the given unit type.

As last attribute, cost $c_U^{(i)}$ finally specifies the price which a producer demands or a consumer is willing to pay for a unit of electrical energy. As already discussed in Section 4.1.2, the assessment of an optimal cost value depends on the trading strategy of an agent because it has an impact on the individual share which the agent receives in the course of payoff distribution.

In order to give a concluding example using the initially specified use case, in the course of a formation process the considered producer might join a coalition and contribute an electricity amount of 4 kWh which is delivered by its controlled CHP plant with a

[8] The root mean square error is generally defined as $RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\widehat{y_i} - y_i)^2}$, with $\widehat{y_i}$ being a predicted and y_i being an observed value.

potential relative deviation of 0.05 (i.e. 0.2 kWh) at cost 0.1 €/kWh in a specific product interval $t_{pr}^{(i)}$, i.e. $con_{a,p}(t_{pr}^{(i)}) = (4\,\text{kWh}, 0.05, 0.1\,€/\text{kWh})$.
Aggregating the contributions of participating agents results in the cumulative contribution $con_{C,p}(t_{pr}^{(i)}) = (e_C^{(i)}, err_C^{(i)}, c_C^{(i)})$ of a coalition C. However, in order to obtain reasonable results, the different attributes have to be aggregated in an appropriate way. With regard to the electricity amount, the quantities of the individual contributions can simply be added up to the total amount of the coalition. I.e., given that $C = \{a_1, \ldots, a_n\}$ with $U_C = \{U_1, \ldots, U_n\}$, the cumulative electricity amount in a product interval $t_{pr}^{(i)}$ is calculated as

$$e_C^{(i)} = \sum_{U \in U_C} e_U^{(i)}. \tag{4.27}$$

Depending on the type of the participating units, the cumulative amount is thus either a positive or a negative value.
Considering the second attribute, the error of a cumulative contribution is defined similar to the unit-related error, i.e. it is specified as the total relative deviation from the aggregated electricity amount which is provided or consumed by a coalition. More precisely, in a product interval $t_{pr}^{(i)}$, the cumulative error is calculated as

$$err_C^{(i)} = \frac{\sqrt{\sum_{U \in U_C} MSE_U^{(i)}}}{|e_C^{(i)}|} = \frac{\sqrt{\sum_{U \in U_C} (err_U^{(i)} \cdot |e_U^{(i)}|)^2}}{|e_C^{(i)}|}. \tag{4.28}$$

The above formula extracts the root from the summed mean squared errors of the units and puts it into relation with the provided or consumed electricity amount of the coalition. In contrast to Equation 4.26, the numerator of the fraction does not calculate the absolute deviation in terms of the $RMSE$ but the root of the sum of the individual $MSEs$. As the square root function $f(x) = \sqrt{x}$ does not grow linearly with the argument x, this approach models the effect that, within a coalition, deviations from prognosticated electricity amounts partially compensate each other. To illustrate, the line plot in Figure 4.10 shows the absolute error $\sqrt{\sum_{U \in U_C} MSE_U}$ as function of the coalition size under the assumption that the contribution of each member specifies an error of 0.1 kWh. As can be seen from the curve, the slope decreases with a growing coalition size as the degree of compensation increases the more members participate. Note that this model assumes that the errors of the different units are uncorrelated. While this might not be the case if some units are based on the same technology or located in the same geographical region, we assume that in the context of our considered problem such correlations are not significant as coalitions are typically large enough in order to entail a sufficient technological diversification and spatial distribution. However, the development of a more sophisticated error model may be addressed by future work (see Section 6.2).

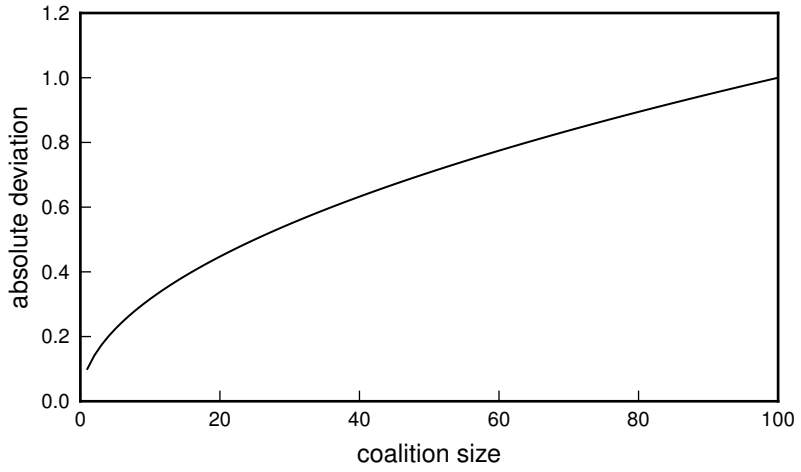


Figure 4.10: Absolute deviation from the cumulative electricity amount.

Finally, the cost of a cumulative contribution is defined in terms of the price which is demanded or paid by a coalition per unit electrical energy. The corresponding formula is given by

$$c_C^{(i)} = \frac{\sum_{U \in U_C} c_U^{(i)} \cdot e_U^{(i)}}{|e_C^{(i)}|}. \tag{4.29}$$

I.e., the cumulative cost is determined by calculating the total energy cost of the individual contributions and then dividing the result by the absolute value of the provided or consumed cumulative electricity amount. Along with the previously specified attributes, this definition finally allows to asses the actual value of a coalition.

4.3.2.2 Coalition Value

In the course of coalition formation, agents have to assess the value of a coalition in order to decide if a potential regrouping increases the value of the global coalition structure in the sense of optimization problem 4.23. In accordance with the latter, a regrouping $(\widehat{C}_{old}, \widehat{C}_{new})$ is generally considered beneficial if

$$\overline{v(C_{new})} > \overline{v(C_{old})}, \tag{4.30}$$

i.e. if the mean coalition value of the resulting coalitions exceeds the one of the original coalitions. As described in the following section, this condition allows agents to optimize global value in the course of the decentralized interactions without having to maintain a complete model of their environment. Moreover, it ensures that a regrouping always leads to an improvement of global value.

As the goal of a coalition C is given by the trade of its target product p_{tar}, the most reasonable way to specify its worth is to evaluate its ability to fulfill the very same. Taking into account the definitions of the previous section, this can be achieved by identifying how well the values of its cumulative contribution $con_{C,p_{tar}}$ approximate the corresponding target values of p_{tar}. Accordingly, the attributes of a target product are generally defined in terms of the cumulative values which are to be met by a final coalition. It is then optimally fulfilled if

$$\forall t_{pr}^{(i)} \in T_{pr}^{(p_{tar})} : con_{C,p_{tar}}(t_{pr}^{(i)}) = (e_C^{(i)}, err_C^{(i)}, c_C^{(i)}) = (e_{p_{tar}}^{(i)}, err_{p_{tar}}^{(i)}, c_{p_{tar}}^{(i)}) = p_{tar}(t_{pr}^{(i)}). \quad (4.31)$$

According to this idea, the approximation of a summed contribution to a target product is defined in the sense of the following attribute-related formulas:

$$e_{C,f}^{(i)} = \varrho(1 - \frac{e_C^{(i)}}{e_{tar}^{(i)}}), \quad (4.32)$$

$$err_{C,f}^{(i)} = \varrho(err_{tar}^{(i)} - err_C^{(i)}), \quad (4.33)$$

$$c_{C,f}^{(i)} = \varrho(c_{tar}^{(i)} - c_C^{(i)}), \quad (4.34)$$

with

$$\varrho : \mathbb{R} \rightarrow]0, 1], \quad \varrho(x) = \frac{1}{1 + |3 \cdot x|}. \quad (4.35)$$

The approximation values $e_{C,f}^{(i)}$, $err_{C,f}^{(i)}$, and $c_{C,f}^{(i)}$ indicate the degree of fulfillment of target electricity amount $e_{tar}^{(i)}$, target error $err_{tar}^{(i)}$, and target cost $c_{tar}^{(i)}$, respectively. They are assessed by first calculating the actual approximations to the target values and then standardizing the outcomes by mapping them to the interval $]0, 1]$ using function ϱ. More precisely, approximation value $e_{C,f}^{(i)}$ is determined by subtracting the achieved percentage of the pursued target electricity amount from 1 and then applying ϱ, whereas $err_{C,f}^{(i)}$ and $c_{C,f}^{(i)}$ are assessed by subtracting the cumulative values from the corresponding target values before standardization. As can be seen from the line plot in Figure 4.11, standardization function ϱ attains its maximum at $x = 0$ in which case the pursued target value is exactly met. As the graph is symmetric about the x-axis, positive and negative deviations from the latter are rated equally. Assuming for instance a target electricity amount of 100 kWh, a cumulative value of 98 kWh is considered as disadvantageous as a value of 102 kWh. Moreover, the non-linear curve promotes the formation of value maximizing coalitions (and thus the fulfillment of their target products) as impacts of changes in x on function value $\varrho(x)$ are the higher, the better the approximation to the target value. For example, assuming again a target electricity amount of 100 kWh, adding a quantity of 10 kWh to a cumulative amount of 80 kWh would lead to a bigger increase in $\varrho(x)$ than adding the same quantity to a cumulative amount of 20 kWh. In this sense, Equation 4.32-4.34 define appropriate arguments being values of similar order of magnitude which do not significantly underrate a specific attribute meaning that changes in the arguments lead to similar changes in the function values.

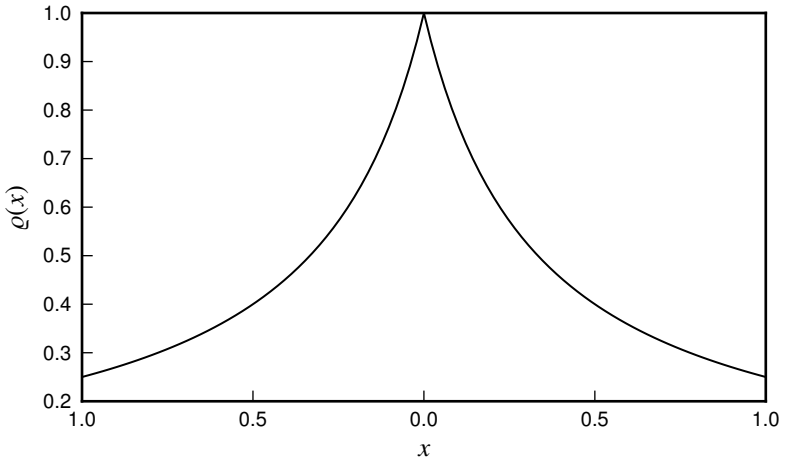


Figure 4.11: Standardization function ϱ.

Based on the above definitions, we are now able to reasonably assess the worth of a coalition. To this end, let $attr = \{e, err, c\}$ be the set of the considered attributes electricity amount, error, and cost. The coalition value of a coalition C striving for the fulfillment of a target product p_{tar} is then defined as a function $v : \widetilde{C} \rightarrow [0, 1]$, with

$$v(C) = \begin{cases} w_e \cdot \overline{e_{C,f}} + w_{err} \cdot \overline{err_{C,f}} + w_c \cdot \overline{c_{C,f}} & \text{if } \overline{e_{C,f}} \neq 0, \\ 0 & \text{else,} \end{cases} \qquad \sum_{i \in attr} w_i = 1, \tag{4.36}$$

where the variables w_i represent weights used to prioritize the different attributes and $\overline{e_{C,f}}$, $\overline{err_{C,f}}$, and $\overline{c_{C,f}}$ are the mean degrees of fulfillment calculated over all product intervals $t_{pr}^{(i)} \in T_{pr}^{(p_{tar})}$. Thus, a coalition is rated according to its ability to fulfill the interval-related target values of the pursued product on average, where its value is set to 0 if no electricity can be provided or consumed within the product horizon at all. As the weights of the attributes are required to sum up to 1 and each approximation value is restricted to the interval $]0, 1]$, $v(C)$ is a standardized value as well allowing agents to reasonably compare the worth of different regroupings.

While Equation 4.36 determines the general value of a coalition, in the course of the formation process agents additionally apply an ad hoc approach for the assessment of a dynamic coalition value $\widetilde{v(C)}$ which further supports the formation of value maximizing coalitions. The underlying idea is to adjust the weights according to the current degrees of fulfillment and put emphasis on the attribute with the lowest mean approximation value while attaching less importance to those showing better results. This way, the optimization of the worst attribute is promoted as a change in the corresponding cumulative value has an increased impact on the coalition value. In order to calculate dynamic

coalition value $\widetilde{v(C)}$, we define an interval $[\widetilde{w}_{i,min}, \widetilde{w}_{i,max}]$ for each attribute $i \in attr$ which specifies the range for a corresponding dynamic weight $\widetilde{w}_i$ replacing its static counterpart w_i in Equation 4.36. Assuming that $\overline{x_{C,f}} < \overline{y_{C,f}} < \overline{z_{C,f}}$, the dynamic weights for the different attributes are then calculated according to the following equations:

$$\widetilde{w}_x = \widetilde{w}_{x,max}, \tag{4.37}$$

$$\widetilde{w}_z = \widetilde{w}_{z,min}, \tag{4.38}$$

$$\widetilde{w}_y = 1 - \widetilde{w}_x - \widetilde{w}_z. \tag{4.39}$$

I.e., the dynamic weights of the attributes with the lowest and highest mean degree of fulfillment are set to the defined maximum and minimum, respectively, while the third weight is calculated by subtracting the former two from the required final sum of one. In order to ensure that the weights in fact add up to one, the defined intervals have to meet the constraint that

$$\forall x, y, z \in attr \text{ with } x \neq y \neq z : \widetilde{w}_{x,max} + \widetilde{w}_{y,min} + \widetilde{w}_{z,min} \leq 1 \leq \widetilde{w}_{x,max} + \widetilde{w}_{y,max} + \widetilde{w}_{z,min}. \tag{4.40}$$

As the approach generally assumes the existence of a unique maximum and minimum approximation value, it is required to define a strict preference relation $\succ$ on the set of attributes *attr* enabling agents to determine dynamic weights even in case values are equal. Because the most pivotal attribute of a product is given by the amount of electricity, we specify that $e \succ err \succ c$ which makes the electricity amount being handled as most important and the error as second most important attribute. In case of total equality (i.e. $\overline{e_{C,f}} = \overline{err_{C,f}} = \overline{c_{C,f}}$), an agent then sets $\widetilde{w}_e$ to its maximum, $\widetilde{w}_c$ to its minimum, and determines $\widetilde{w}_{err}$ according to Equation 4.39. As the goal of coalition formation is given by the maximization of global value, it is reasonable to prescribe the same weights w_i and intervals $[\widetilde{w}_{i,min}, \widetilde{w}_{i,max}]$ for all participating agents. This can be achieved through the definition of a regulatory hard constraint hc_a.

4.3.2.3 Interaction, Optimization, and Deadlock/Livelock Prevention

Based on the approach to assess the value of a coalition as described in the previous section, agents conduct the activity of coalition formation in order to cooperate with each other and finally fulfill their pursued target products. Generally, they start as singleton coalitions and iteratively apply an interaction protocol which allows them to heuristically evaluate potential regroupings and reorganize themselves into optimized coalitions in a fully decentralized and self-organized fashion. As the protocol forms the basis for the process flow of the activity, in the following we first provide its detailed specification along with the algorithms applied for the identification of value optimizing regroupings. Afterwards, we discuss situations which cause deadlocks and livelocks in the course of the negotiations and describe mechanisms for their prevention.

The interaction protocol which is used by agents for the decentralized formation process is depicted by the sequence diagrams in Figure 4.12 and 4.13. Its specification is based on the contract net protocol, a well-established approach for cooperative problem solving

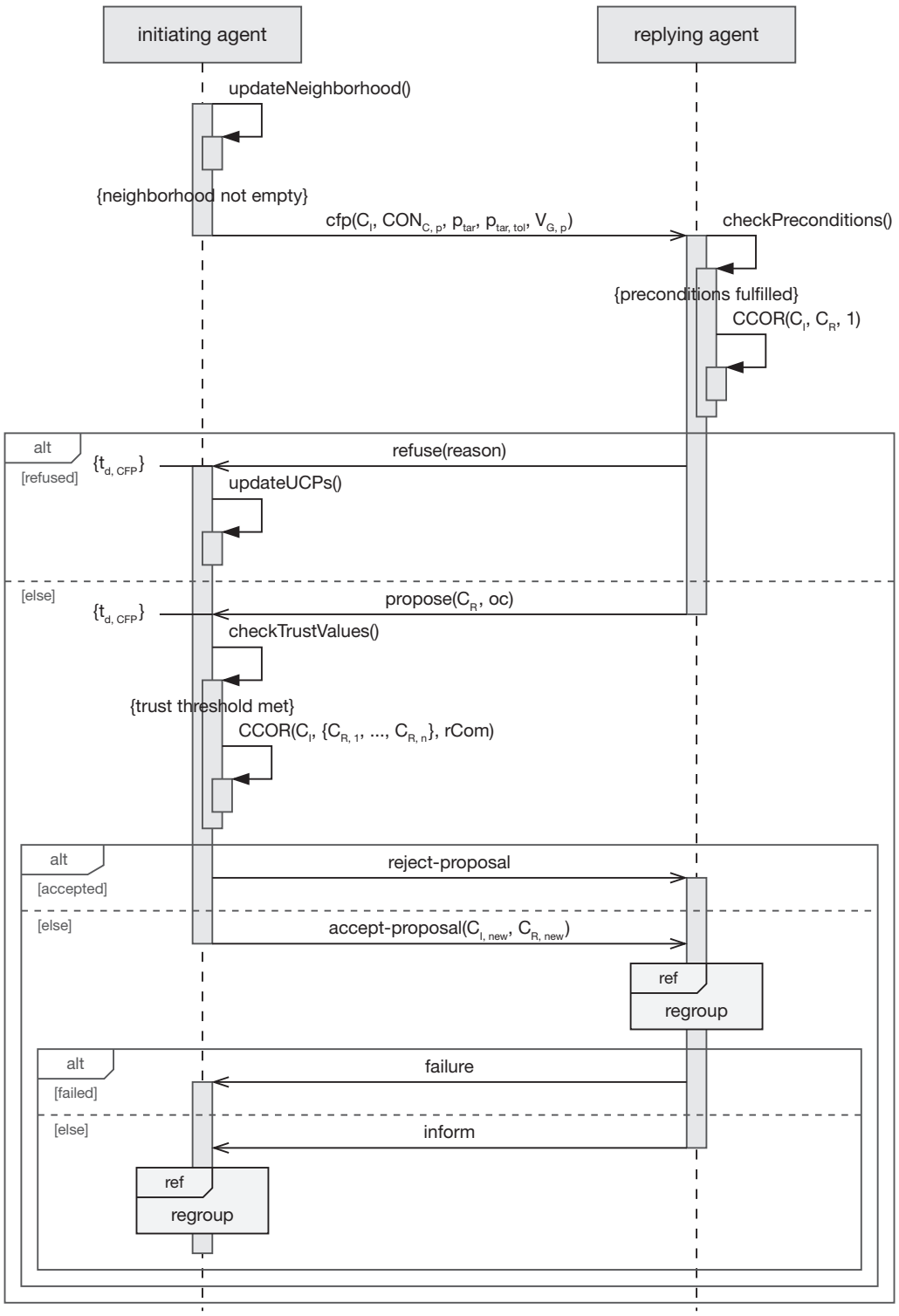


Figure 4.12: Interaction protocol for coalition formation.

which was standardized by the Foundation for Intelligent Physical Agents as described in Section 2.2.2.2. For the context of this thesis, we adapt the FIPA standard to our needs and extend it by additional interactions which allow participants to regroup coalitions and maintain a consistent data model.
In the course of coalition formation, an agent can generally take on one of two different roles which determines its behavior throughout the negotiations. First, it can act as an initiator which sends formation requests to other participants and replies to requests received from other agents. Alternatively, it can assume the role of a responder which replies to received requests only. Equipping initiators with both capabilities provides the advantage to achieve optimized solutions more quickly as agents join faster into bigger coalitions and to investigate larger parts of the search space as more regroupings are examined. In the course of a formation process, agents are generally able to switch roles if specific conditions are met. These are described in more detail in the context of the process flow discussed in the following section.
As indicated by the lifelines in Figure 4.12, a regrouping of coalitions is generally negotiated between an initiating agent a_I and a replying agent a_R, where the former always represents an initiator and the latter is either given by an initiator or a responder. In the following, we refer to the corresponding coalitions as initiating coalition C_I and replying coalition C_R and further distinguish between a replying initiator coalition C_{RI} and a replying responder coalition C_{RR} where necessary.
As first action of the protocol, an initiating agent a_I determines the receivers of its intended formation request by temporarily removing those agents from its neighborhood which are either

- members of its own coalition,
- participants not satisfying a specified minimum trust threshold $trust_{min,a_I}$, or
- participants being considered as unlikely cooperation partners ucp.

Excluding all agents a_j which trust values $trust((t, a_I, a_j))$ do not meet its defined standard allows a_I to prevent unfavourable cooperations in advance. Further removing all participants which are assumed to offer no prospect of a promising negotiation additionally enables a reduction of communication cost through a minimization of unnecessary requests. As described in more detail below, unlikely cooperation partners are identified by means of standardized replies which are sent by agents if specific conditions are met.
In case the resulting neighborhood is not empty, agent a_I sends a call for proposal (CFP) to the comprised neighbors, where the payload of the message is given by its coalition C_I and the individual contributions of the members $CON_{C_I,p_{tar}}$ as well as the pursued target product p_{tar} and the corresponding tolerance bands $p_{tar,tol}$. If p_{tar} is a localized product, a_I also adds the required regional information in the form of the grid nodes $V_{G,p_{tar}}$ which define the grid section in which the product is to be physically fulfilled. As further message parameter, it finally attaches a deadline $t_{d,CFP}$ specifying the time by which it handles replies to the call at the latest. As described at the end of this section, this measure allows to prevent deadlocks throughout the negotiations.

Having received a CFP, a replying agent a_R first evaluates the preconditions which have to be fulfilled for a reasonable evaluation of a member transfer between the transmitted coalition and its own. More precisely, if a_R is an initiator, it first ensures mutual exclusion by checking if a reply is in line with the mechanism for livelock prevention described in more detail below. Second, it verifies if the target products are compliant which is given if both negotiation partners either strive for the same target values and tolerance bands or the values of the initiating agent are adoptable by the replying agent. In case a_I pursues a localized product, it is additionally checked if all members of coalition C_R are connected to a grid node of the specified region $V_{G,p_{tar}}$, i.e. if $\forall a_i \in C_R : gn(u(a_i)) \in V_{G,p_{tar}}$. As last precondition, a_R verifies if all members of coalition C_I fulfill its own minimum trust threshold $trust_{min,a_R}$.

In case both preconditions are fulfilled, agent a_R tries to identify a potential regrouping $(\widehat{C}_{old}, \widehat{C}_{new})$ which optimizes global value in terms of regrouping rule 4.30. To this end, it applies a heuristic termed CCOR[9] – Combined COalition Regrouping – which allows for a member transfer between an initiating coalition $C_{I,old}$ and a set of replying coalitions $\widetilde{C}_{R,old}$ (see Algorithm 4.3). This is generally achieved by iterating over the latter and executing in each cycle a regrouping method which implements a systematic member transfer between the current instance of the initiating coalition $C'_{I,old}$ and the currently considered replying coalition $C'_{R,old} \in \widetilde{C}_{R,old}$ (see Algorithm 4.2). The method basically consists of two complementary steps in which members are first transferred one-way from the replying to the initiating coalition in order to cumulate potentials and then exchanged between both coalitions in order to improve the approximation to the values of the pursued target product. By iterating over $\widetilde{C}_{R,old}$, CCOR thus successively improves global value until a specified number of replying coalitions has been processed. This also means that when applied for the regrouping of a single initiating and replying coalition as at this point of the negotiation process, the output of CCOR equals the one of the integrated regrouping method. In order to facilitate discussion, we thus first discuss the functionality of the latter and defer the description of Algorithm 4.3 to the point when it is again used by the initiating agent later in the process.

The input parameters of regrouping method 4.2 are given by the original coalitions $C_{I,old}$ and $C_{R,old}$ as well as a further set of agents $C_{I,valid}$ which comprises those members of $C_{I,old}$ being generally transferable to $C_{R,old}$. $C_{I,valid}$ only differs from $C_{I,old}$ if CCOR is applied for a regrouping involving more than one replying coalition as described in more detail below. When the algorithm is executed by the replying agent at this point of the negotiations, it thus always holds that $C_{I,valid} = C_{I,old}$, meaning that all members of $C_{I,old}$ are transferable to $C_{R,old}$. After termination, the output of the regrouping method is given by the result sets $\widehat{C}_{I,new}$ and $\widehat{C}_{R,new}$ comprising the optimized initiator and responder coalition, respectively[10]. In this regard, it generally holds that $0 \leq |\widehat{C}_{R,new}| \leq |\widehat{C}_{I,new}| \leq 1$, i.e. the regrouped coalitions result from the original ones either through partial reorganization (in which case $|\widehat{C}_{R,new}| = |\widehat{C}_{I,new}| = 1$) or a total

[9] pronounced as [si:kɔ:r]

[10] With respect to Definition 3.12 (regrouping), the input and output parameters refer to the specified notion $(\widehat{C}_{old}, \widehat{C}_{new})$ in the sense that $\widehat{C}_{old} = \{C_{I,old}, C_{R,old}\}$ and $\widehat{C}_{new} = \widehat{C}_{I,new} \cup \widehat{C}_{R,new}$.

Algorithm 4.2: *regroupCoalitions*$(C_{I,old}, C_{R,old}, C_{I,valid})$

input : $C_{I,old}$, $C_{R,old}$, $C_{I,valid}$
output: $\widehat{C}_{I,new}$, $\widehat{C}_{R,new}$

$\widehat{C}_{I,new}, \widehat{C}_{R,new} \leftarrow \emptyset, \emptyset$ // $\Leftrightarrow \widehat{C}_{I,new} \leftarrow \emptyset; \widehat{C}_{R,new} \leftarrow \emptyset$
$C_{I,new}, C_{R,new} \leftarrow C_{I,old}, C_{R,old}$
$C'_{I,new}, C'_{R,new} \leftarrow C_{I,old}, C_{R,old}$
foreach $a_r \in C_{R,old}$ **do**
  $C'_{R,new} \leftarrow C'_{R,new} \backslash \{a_r\}$
  $C'_{I,new} \leftarrow C'_{I,new} \cup \{a_r\}$
  if $(\overline{v(C'_{new})} > \overline{v(C_{new})}) \wedge (stillSuc(C_{I,old}, C'_{I,new}, C_{R,old}, C'_{R,new}))$ **then**
    $C_{I,new}, C_{R,new} \leftarrow C'_{I,new}, C'_{R,new}$
  end
end
if $(\neg succ(C_{I,new})) \wedge (C_{R,new} \neq \emptyset) \wedge (|C_{I,new}| \neq 1 \wedge |C_{R,new}| \neq 1)$ **then**
  $C'_{I,new}, C'_{R,new} \leftarrow C_{I,new}, C_{R,new}$
  $C''_{R,new} \leftarrow C_{R,new}$
  foreach $a_i \in C_{I,valid}$ **do**
    $C'_{I,new} \leftarrow C'_{I,new} \backslash \{a_i\}$
    $C'_{R,new} \leftarrow C'_{R,new} \cup \{a_i\}$
    if $(\overline{v(C'_{new})} > \overline{v(C_{new})}) \wedge (stillSuc(C_{I,old}, C'_{I,new}, C_{R,old}, C'_{R,new}))$ **then**
      $C_{I,new}, C_{R,new} \leftarrow C'_{I,new}, C'_{R,new}$
    end
    foreach $a_r \in C''_{R,new}$ **do**
      $C'_{I,new} \leftarrow C'_{I,new} \cup \{a_r\}$
      $C'_{R,new} \leftarrow C'_{R,new} \backslash \{a_r\}$
      if $(\overline{v(C'_{new})} > \overline{v(C_{new})}) \wedge (stillSuc(C_{I,old}, C'_{I,new}, C_{R,old}, C'_{R,new}))$ **then**
        $C_{I,new}, C_{R,new} \leftarrow C'_{I,new}, C'_{R,new}$
      end
      $C'_{I,new} \leftarrow C'_{I,new} \backslash \{a_r\}$
      $C'_{R,new} \leftarrow C'_{R,new} \cup \{a_r\}$
    end
    $C'_{I,new} \leftarrow C'_{I,new} \cup \{a_i\}$
    $C'_{R,new} \leftarrow C'_{R,new} \backslash \{a_i\}$
  end
end
checkAndUpdateRepresentatives$(C_{I,old}, C_{I,new}, C_{R,old}, C_{R,new})$
addRegroupedCoalitions$(C_{I,new}, \widehat{C}_{I,new}, C_{R,new}, \widehat{C}_{R,new})$
return $\widehat{C}_{I,new}, \widehat{C}_{R,new}$

merge (in which case $|\widehat{C}_{R,new}| = 0$ and $|\widehat{C}_{I,new}| = 1$). If both sets are empty, no value optimizing regrouping could be achieved throughout execution.
Considering its implemented functionality, the method starts by initializing the output variables $\widehat{C}_{I,new}$ and $\widehat{C}_{R,new}$ with the empty set and the finally regrouped coalitions $C_{I,new}$ and $C_{R,new}$ as well as the temporary variables $C'_{I,new}$ and $C'_{R,new}$ with the corresponding input coalitions (line 1-3). Throughout execution, $C_{I,new}$ and $C_{R,new}$ always store the best solution which has been identified so far. Next, the above outlined optimization process begins in which agents are first transferred one-way from the replying to the initiating coalition in order to aggregate technical potentials (line 4-10) and then exchanged between both coalitions in order to approximate the given target values more accurately (line 11-32). In the course of execution, a fulfillment of target products is generally preserved in the sense that only those regroupings are considered valid which do not compromise a prior success of the original coalitions[11]. The first part of the optimization process is implemented by the foreach loop in line 4-10 which iterates over the original members of the replying coalition and passes them one by one to the initiating coalition while examining the benefit of the regroupings. When the improvement of the mean coalition value is evaluated in line 7, function *stillSuc* additionally checks if product fulfillment is preserved, i.e. if $C'_{I,new}$ and $C'_{R,new}$ still fulfill their target products if $C_{I,old}$ and $C_{R,old}$ already did. In case both conditions evaluate to true, the temporary coalitions are saved as currently best solution.
Having transferred all members to the initiating coalition, the method executes the second part of the optimization process and first checks the general preconditions for a member exchange by evaluating if the currently best initiator coalition $C_{I,new}$ is still unsuccessful, the currently best replying coalition $C_{R,new}$ is not empty, and both coalitions are not singletons (line 11). If all conditions are fulfilled, it sets the temporary variables $C'_{I,new}$ and $C'_{R,new}$ to the best coalitions found so far and additionally stores the state of $C_{R,new}$ in another temporary variable $C''_{R,old}$. The following nested foreach loops then exchange each transferable member of the initiating coalition by each member of the replying coalition. More precisely, the first loop starts by iterating over the valid initiator members $C_{I,valid}$ and transferring the currently considered agent a_i from coalition $C'_{I,new}$ to $C'_{R,new}$ (line 15-16). If this regrouping already improves global value and preserves all prior product fulfillments, it is stored as currently best solution. This first check is particularly conducted in order to allow initiating coalitions to approximate their target electricity amounts to a higher degree if they currently exceed their pursued values. The second foreach loop then iterates over the members of temporary coalition $C''_{R,old}$ (i.e. of the best replying coalition as identified after the first optimization part) and transfers them back and forth between the initiating and the replying coalition, in between evaluating if an exchange improves global value under the given constraints (line 20-28). Afterwards, the outer loop passes agent a_i back to $C'_{I,new}$ and starts the iteration for the next member until all agents were exchanged against each other. In line 33,

[11] Note that in spite of product fulfillment, a successful coalition might still provide optimization potential because of the flexibility provided by the tolerance bands $p_{tar,tol}$ associated with the different attributes of the target product.

method *checkAndUpdateRepresentatives* then checks if the representatives of the original coalitions were transferred in the course of the optimization process. If this is true in one of the cases, a new one is determined by randomly assigning the task to an alternative member of the new coalition. Method *addRegroupedCoalitions* finally adds the coalitions $C_{I,new}$ and $C_{R,new}$ to their corresponding result set given that they are not equal to the original coalitions $C_{I,old}$ and $C_{R,old}$ and that they comprise at least one agent. The regrouping method then terminates by returning $\widehat{C}_{I,new}$ and $\widehat{C}_{R,new}$ as output, where both sets are empty if no optimization could be achieved. Because the replying agent only examined a single regrouping of the initiating coalition and its own, these are generally equal to the result sets which are finally returned by the calling CCOR algorithm (i.e. $\widehat{C}_{I,com}$ and $\widehat{C}_{R,com}$) as described above.
Depending on the outcome of the evaluation process, the replying agent proceeds with the negotiations by sending either a refusal or a proposal as answer to the received CFP (see Figure 4.12). A refusal is sent if one of the initial preconditions was not fulfilled or CCOR did not identify a regrouping yielding an increased global value. In order to inform the initiating agent about the specific reason, the agent includes one of the predefined tokens

<reason> ::= “mutex” | “product” | “trust” | “optimization”

as payload of the message, where the first three keywords indicate a non-fulfillment of the corresponding preconditions while the last one signals a current lack of optimization potential. Contrary, if all preconditions were met and an optimizing regrouping was identified, a_R sends a proposal to a_I including its own coalition C_R as payload. Moreover, it adds another parameter *oc* indicating the number of additional optimization cycles which it has conducted given that the coalition is already successful. As described in the next section, this mechanism is applied by all representatives in order to further improve global value and approximate the values of the pursued target products to a higher degree. Because in the course of a regrouping representatives may change, a_R adds the number as further contents which is then taken over by a successor.
Agent a_I starts with the processing of all replies if it has either obtained a response from all recipients of its call or the specified deadline has expired[12]. When evaluating the received messages, it first examines each refusal with regard to its specified reason. If the latter indicates a non-compliance of the target products or a lack of trustworthiness, the sender is categorized as unlikely cooperation partner *ucp* and added to a corresponding list which is then used in the course of further formation attempts in order to exclude the comprised agents from the neighborhood as described above. Having handled all refusals, a_I next processes the received proposals and first checks if the members of the transmitted coalitions C_R meet its minimum trust threshold $trust_{min,a_I}$. Afterwards, it applies the CCOR algorithm for a combined regrouping involving all replying coalitions which passed the test. As specified by Algorithm 4.3, the input of the heuristic is given by the original initiating coalition $C_{I,old}$, the set of replying coalitions

[12] Under specific circumstances, initiators deviate from this behavior in order to prevent livelocks as described later in this section.

Algorithm 4.3: Combined Coalition Regrouping

input : $C_{I,old}$, $\widetilde{C}_{R,old}$, $rCom$
output: $\widehat{C}_{I,com}$, $\widehat{C}_{R,com}$

1 $\widehat{C}_{I,com}, \widehat{C}_{R,com} \leftarrow \emptyset, \emptyset$
2 $C'_{I,old} \leftarrow C_{I,old}$
3 **foreach** $C'_{R,old} \in \widetilde{C}_{R,old}$ **do**
4 $\quad C_{I,valid} \leftarrow C'_{I,old} \cap C_{I,old}$
5 $\quad \widehat{C}_{I,new}, \widehat{C}_{R,new} \leftarrow regroupCoalitions(C'_{I,old}, C'_{R,old}, C_{I,valid})$
6 $\quad$ **if** $\widehat{C}_{I,new} \neq \emptyset$ **then**
7 $\quad\quad C'_{I,old} \leftarrow C_{I,new}$
8 $\quad\quad \widehat{C}_{I,com} \leftarrow \widehat{C}_{I,new}$
9 $\quad\quad \widehat{C}_{R,com} \leftarrow \widehat{C}_{R,com} \cup \widehat{C}_{R,new}$
10 $\quad\quad$ **if** $|\widehat{C}_{R,com}| = rCom$ **then**
11 $\quad\quad\quad$ **break**
12 $\quad\quad$ **end**
13 $\quad$ **end**
14 **end**
15 **return** $\widehat{C}_{I,com}$, $\widehat{C}_{R,com}$

$\widetilde{C}_{R,old} = \{C_{R,old,1}, \ldots, C_{R,old,n}\}$, as well as a further parameter $rCom$ which allows initiating agents to vary the maximum number of accepted proposals by restricting the amount of replying coalitions being part of the final regrouping. As output, the algorithm finally returns the result sets $\widehat{C}_{I,com}$ and $\widehat{C}_{R,com}$ which comprise the regrouped initiating and replying coalitions, where it generally holds that $|\widehat{C}_{I,com}| = 1$ and $0 \leq |\widehat{C}_{R,com}| \leq rCom$. With regard to the implemented functionality, the code starts by initializing the output variables with the empty set and setting temporary variable $C'_{I,old}$ to input coalition $C_{I,old}$ (line 1-2). In the course of the optimization process, $C'_{I,old}$ always stores the best solution found so far. The following foreach loop then conducts the combined regrouping by iterating over the set of replying coalitions and repeatedly calling regrouping method 4.2, where $C'_{I,old}$, the currently considered replying coalition $C'_{R,old}$, and the previously determined set of generally transferable initiator members $C_{I,valid}$ are passed as arguments. In each iteration, the latter is calculated as intersection of optimized coalition $C'_{I,old}$ (potentially comprising replying members which were already transferred to $C'_{I,old}$ in prior iterations) and original coalition $C_{I,old}$ (line 4). Hence, $C_{I,valid}$ only includes the remaining initiator members which prevents a further transfer of transferred replying members to other coalitions. In line 6, the algorithm then evaluates if the application of the regrouping method led to an improvement of global value by checking if result set $\widehat{C}_{I,new}$ includes an optimized initiating coalition. If this is true, it first stores the latter as input for the next iteration in temporary variable $C'_{I,old}$ and then updates the result sets by

setting $\widehat{C}_{I,com}$ to $\widehat{C}_{I,new}$ and adding the potential replying coalition of $\widehat{C}_{R,new}$ to $\widehat{C}_{R,com}$. Finally, it checks the size of $\widehat{C}_{R,com}$ and exits the loop if the maximum number of accepted replying coalitions has been reached (line 10-12). Having finished the combined regrouping, CCOR terminates by returning the result sets $\widehat{C}_{I,com}$ and $\widehat{C}_{R,com}$ as output. Depending on the optimization potential of the input coalitions, these either contain the regrouped initiating and replying coalitions or are empty if no improvement in global value could be achieved.

Having described its implemented functionality, we next determine the performance of CCOR in terms of the O-notation. To this end, we first assess the running time of regrouping method 4.2 which is iteratively called throughout execution. The variable input parameters which are relevant for the analysis are given by replying coalition $C_{R,old}$ as well as the set of transferable initiator members $C_{I,valid}$, whereas initiating coalition $C_{I,old}$ can be omitted as the method only considers those initiator members for a transfer which are part of the latter. Thus, for the following discussion let $m = |C_{I,valid}|$ and $n = |C_{R,old}|$. The first code segment which has a relevant impact on the performance is given by the foreach loop in line 4-10 iterating over the members of the original replying coalition $C_{R,old}$. As all comprised statements run in constant time, the running time of the loop is $O(m)$. Afterwards, the nested foreach loops in line 14-31 and 20-28 iterate over the set of transferable initiator members $C_{I,valid}$ and temporary replying coalition $C''_{R,new}$, respectively. Because all included statements again run in constant time and in worst case it holds that $|C''_{R,new}| = |C_{R,old}|$, the running time of the segment is $O(m \cdot n)$. As the remaining code in line 33-35 also runs in constant time, it follows that the total running time of the regrouping method is $O(m + m \cdot n) = O(m \cdot n)$. Given this, it can be seen from Algorithm 4.3 that CCOR calls the method $l = \max(|\widetilde{C}_{R,old}|, rCom)$ times without the execution of any further relevant statements. Hence, the running time of the heuristic is $O(l \cdot m \cdot n)$. Even for large coalition sizes in the order of thousands, CCOR thus provides an efficient approach for the identification of value optimizing regroupings.

Based on the output of CCOR, initiating agent a_I proceeds with the negotiations by rejecting the proposals from all agents which are not part of one of the returned coalitions. Contrary, it sends acceptances to all other representatives for which coalitions a value improving regrouping could be found, where the payload of a corresponding message is given by optimized initiating coalition $C_{I,new}$ and, given that the coalitions do not merge, replying coalition $C_{R,new}$.

When the replying agent receives a positive answer to its proposal, it performs the first part of the actual regrouping which affects all transferred members of its own coalition. Afterwards, it sends a notification to the initiating agent informing it about the failure or success of the action. In the latter case, the initiating agent completes the regrouping by performing all complementing transfer activities. The interactions which are conducted by both agents in the course of the regrouping process are shown by the sequence diagram in Figure 4.13. Generally, the specified actions guarantee a consistent system state by ensuring that after their execution

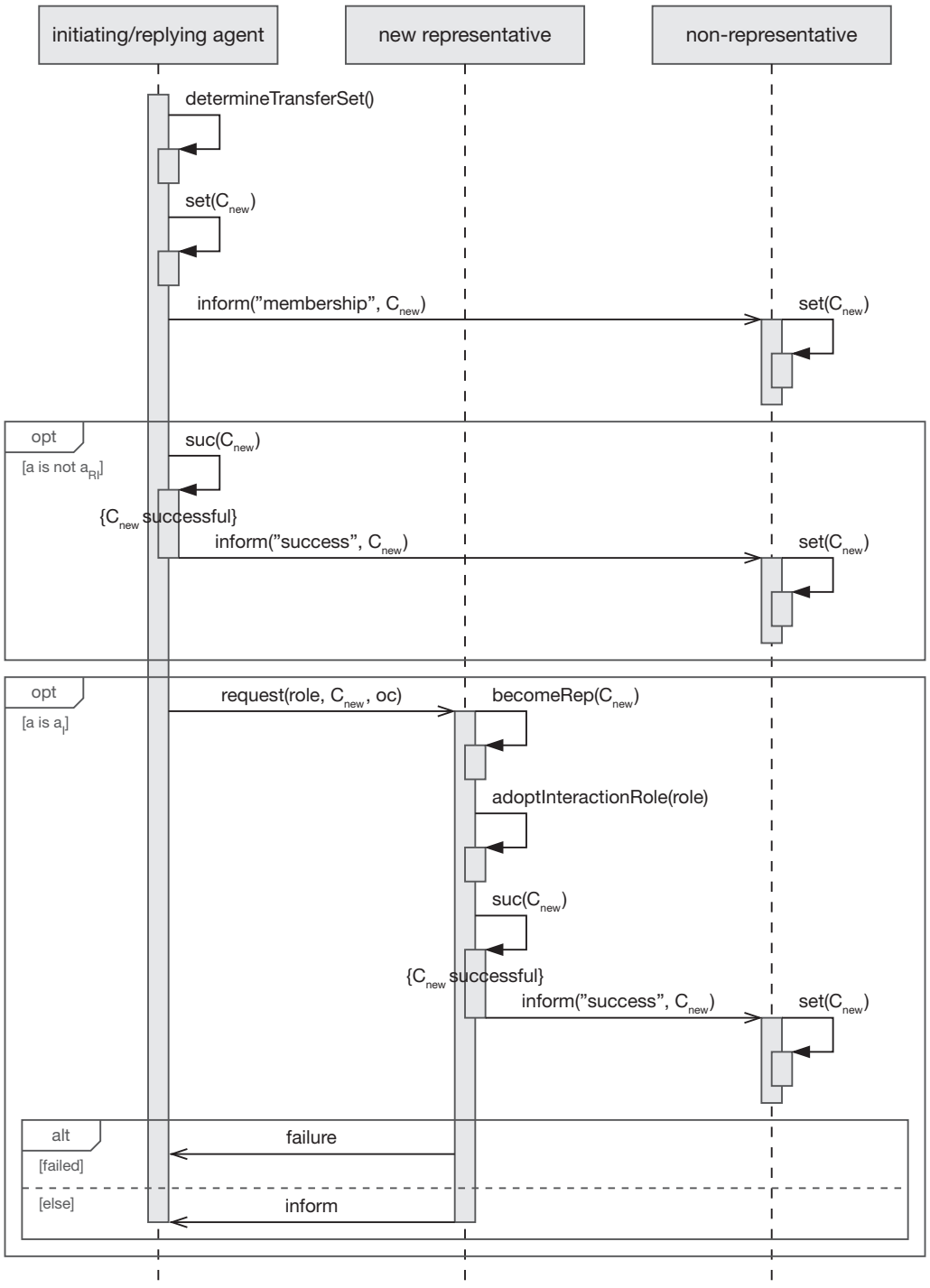


Figure 4.13: Interaction occurrence *regroup* of sequence diagram 4.12.

- all transferred agents have updated their internal data model with regard to their new membership,
- all new representatives have taken over their assigned task and adopted the interaction role of their predecessor, and
- all representatives have evaluated the fulfillment of their pursued target product and informed all members in case the regrouping led to a success of their coalition.

More precisely, the replying agent starts the regrouping by first identifying those members of its original coalition $C_{R,old}$ which have to be transferred to the new initiating coalition $C_{I,new}$, where the corresponding set is defined as $C_{tran} = (C_{R,old} \setminus \{a_R\}) \cap C_{I,new}$. Next, it updates its internal data model with regard to its new coalition C_{new} which either equals $C_{I,new}$ in case the agent switches coalitions as well or $C_{R,new}$ otherwise. The transfer of the identified members is then conducted by sending a message to all agents $a \in C_{tran}$ informing them about their new membership in initiating coalition $C_{I,new}$, where the addressees update their data model accordingly upon receival. If the agent stays representative of its coalition and is not a replying initiator a_{RI} (because in this case it has first to complete its current optimization cycle in the form of another initiation as described in more detail in the next section), it finally evaluates if the regrouping led to a success and the new cumulative contribution meets the values of the pursued target product p_{tar} given the tolerance bands $p_{tar,tol}$ (cf. Definition 3.41). If this is the case, it informs all members about the final coalition which then update their data model accordingly.
The regrouping actions conducted by the initiating agent are generally analogous to those of the replying agent, with the difference that they also comprise a final notification sent to the new representatives if the original ones were transferred in the course of the optimization process. Generally, this task can not be performed by a_R as new initiators would be able to start new negotiations before the current regrouping has been completely finished. The initiating agent starts the process by determining the set of members to be transferred which is defined as $C_{tran} = ((C_{I,old} \setminus \{a_I\}) \cap C_{R,new}) \setminus A_{rep}$, with A_{rep} being the set of new representatives which were determined by CCOR throughout execution. Next, it sets its new coalition and notifies all identified agents with regard to the transfer to replying coalition $C_{R,new}$. Given that it does not switch coalitions itself, the agent then informs all members in case of a success and finally concludes the regrouping process by sending requests for a representative change to all participants which were newly assigned to the task. The payload of the corresponding message comprises the coalition to represent (i.e. $C_{I,new}$ or $C_{R,new}$) as well as one of the predefined tokens

<role> ::= “initiator” | “responder”

indicating the interaction role of the old representative. Moreover, the message includes the number of additional optimization cycles oc which have already been conducted for the transmitted coalition. A recipient responds to the request by taking over the assigned task and adopting the specified role. Afterwards, it evaluates the success of

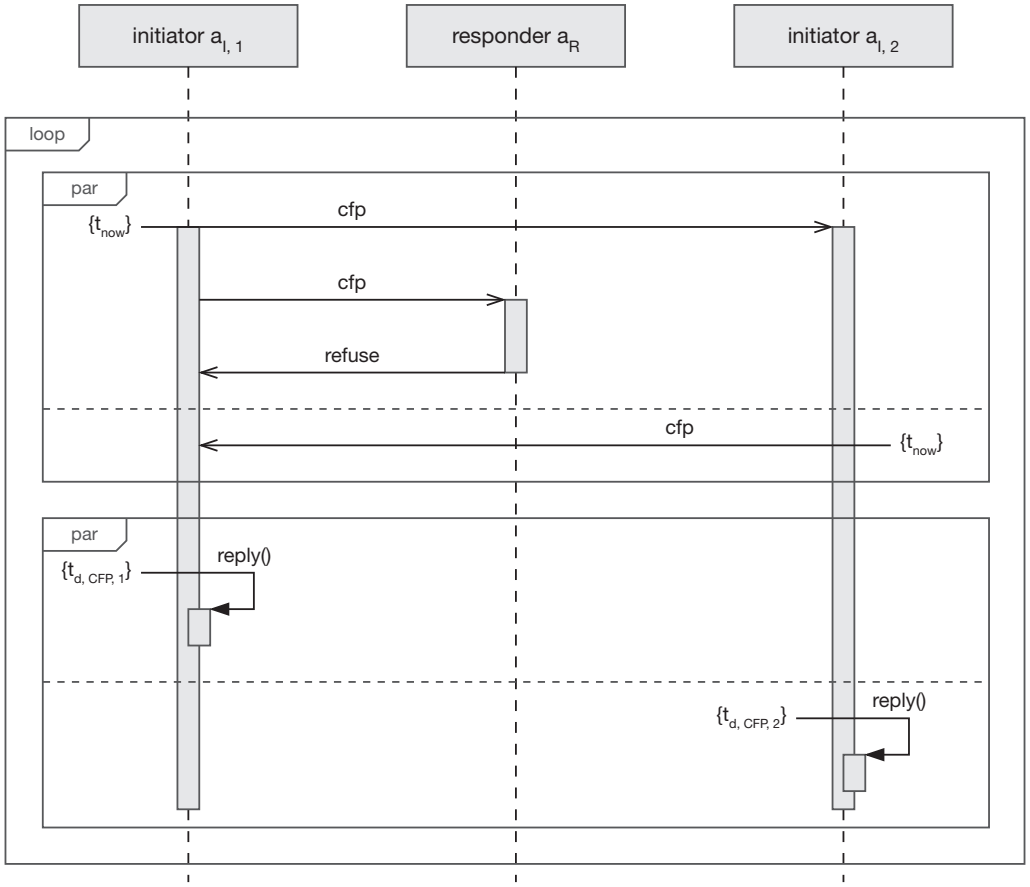


Figure 4.14: Negotiation without livelock prevention.

its new coalition and informs the members according to the outcome. This behavior is consistent with the actions of the initiating and replying agent which only performed the check if they stayed representative of their coalition.
After all required regrouping activities have been finished, interaction protocol 4.12 is finally completed leaving the system in a consistent state. However, as negotiations are carried out in a parallel fashion, there is still the risk that agents run into deadlocks or livelocks throughout the formation process, i.e. situations in which they are either unable to proceed with any further interaction or interact constantly but without any effect. To illustrate, the sequence diagram in Figure 4.14 shows a negotiation involving two initiators $a_{I,1}$ and $a_{I,2}$ as well as a responder a_R. As depicted, at time t_{now} agent $a_{I,1}$ sends a CFP to $a_{I,2}$ and a_R and agent $a_{I,2}$ simultaneously sends a CFP to $a_{I,1}$. If in such situations initiators do not specify a time by which they automatically proceed with the process although not all replies have been received yet, they are stuck in a deadlock

endlessly waiting for remaining answers. Thus, initiating agents attach a deadline $t_{d,CFP}$ to their calls as described in the context of the protocol specification above.
Besides deadlocks, initiators may also end up in livelocks in which case they constantly start new negotiations but their requests have no effect. Suppose for instance that in the considered example agent $a_{I,1}$ and $a_{I,2}$ specify the same deadline $t_{d,CFP,1} = t_{d,CFP,2}$ for deadlock prevention and that responder a_R refuses the call from $a_{I,1}$ as shown in the diagram. Because after expiration of the deadlines both initiators have not obtained a proposal, they proceed by checking for a received CFP with the intention to reply[13]. However, because both deadlines were equal and the mutually sent calls are thus not valid any more, the agents discard the latter and start the next regrouping attempt resulting in the same interaction sequence which hence continues endlessly without effect.
In order to prevent this kind of situations, initiators apply a mechanism which allows them to detect livelocks and synchronize their interactions through tracking of their communication. More specifically, each initiating agent maintains two dedicated lists CFP_{out} and CFP_{in} comprising the addressees of a CFP sent at the beginning of interaction protocol 4.12 and the senders of CFPs which are received during the regrouping attempt. While waiting for replies to a call, an initiator then constantly listens for incoming messages and removes senders of refusals from the first list while adding those of CFPs to the second one. Moreover, after each update it checks CFP_{out} and CFP_{in} for equality in order to determine if it has received a call from all those recipients which have not answered its CFP yet. Because in this case a receival of valid replies is either highly unlikely or – if initiators apply the same deadlines $t_{d,CFP}$ – even impossible, it proceeds by examining the received calls with regard to a globally applied hard constraint guaranteeing mutual exclusion and answers the first message which is consistent with the rule. More specifically, the constraint ensures that two initiators (in our example $a_{I,1}$ and $a_{I,2}$) do not simultaneously reply to mutually sent CFPs and cause an inconsistent system state because of concurrently conducted regroupings. This is achieved by allowing only that agent to reply which identifier lexicographically precedes the one of its negotiation partner. As the identifier is a globally unique constant, the rule guarantees a consistent behavior of all participants[14].
Assuming that this mechanism is applied by the agents in the above example, initiator $a_{I,1}$ then starts the negotiations by sending a CFP to $a_{I,2}$ and a_R and adding both recipients to list CFP_{out}. When receiving the call from $a_{I,2}$, it next updates CFP_{in} and remains waiting for further messages as the list is not equal to CFP_{out}. However, upon receival of the refusal from a_R, it removes the latter from CFP_{out} and replies to the obtained CFP because first the requirement of equality is fulfilled and second its identifier lexicographically precedes the one of agent $a_{I,2}$ (supposing that the variable names represent the IDs). Correspondingly, $a_{I,2}$ remains waiting when receiving the call from $a_{I,1}$ as the

[13] A detailed description of an initiator's behavior after the completion of interaction protocol 4.12 is given in the following section.
[14] If DYCE is applied to a scenario in which total equality of all agents is an issue, the regulatory hard constraint can be altered by simply reversing the condition for different formation intervals $t_{f,p_{tmp}}$.

check for mutual exclusion prevents it to answer. As a result, both agents are finally synchronized again and able to successfully proceed with the negotiations.

4.3.2.4 Process Flow

Having specified the interaction protocol which is used by agents in order to coordinate their actions, we are finally ready to describe the process flow which they run through in the course of coalition formation as well as additional mechanisms which they apply to further improve global value. The corresponding activity diagram and its related guard conditions are given in Figure 4.15 and Table 4.1, respectively. The depicted process comprises all actions which an agent is generally able to carry out in the course of the decentralized interactions, i.e. it can initiate a regrouping, reply to regrouping requests, stand by if it is not available for any negotiations, switch its interaction role, and terminate the activity. As the actual behavior of an agent particularly depends on the interaction role which it selected in the course of neighborhood formation, in the following we first discuss coalition formation from the viewpoint of an initiator and then from that of a responder.

Having determined its neighborhood N, an initiator begins the formation process by starting a new regrouping attempt through application of interaction protocol 4.12. Given that the current formation interval $t_{f,p_{tmp}}$ has not yet expired, its following action then depends on the result of the negotiations from which it can either emerge as representative of its own coalition C_I or as new member (i.e. representative or non-representative) of the replying coalition C_R. If it stays representative of its own coalition and the latter is still not successful, it continues the formation process by checking for a received CFP with the intention to reply (guard condition $gc_{7,1}$). Even in case of a success in which all values of the cumulative contribution lie within the defined tolerance bands, it strives for a further improvement by conducting a maximum of oc_{max} optimization cycles in the form of a corresponding number of additional initiations in order to approximate the pursued target values to a higher degree ($gc_{7,2}$). Correspondingly, after the maximum number of cycles has been completed, the agent finally considers its target product as fulfilled and switches into a standby mode in which it rejects all further regrouping requests ($gc_{6,1}$). The same holds if it became a non-representative member of the new replying coalition C_R ($gc_{6,3}$). However, in this case it can again participate actively in future negotiations as an initiator (gc_{13}) or a responder (gc_{14}) if it was selected as a corresponding representative in the context of a further regrouping. Moreover, the agent can be transferred as new non-representative member to another coalition in which case it updates its data model as described above but remains standing by.

Contrary, if the agent leaves the negotiations as new representative of the replying coalition, its next action depends on the interaction role of its predecessor. If it adopted the role of an initiator, it restarts the process by determining a neighborhood and starting a negotiation given its new coalition C_{RI} (gc_5). In this case, its behavior is generally independent from the success of the latter because even in case the target product is already fulfilled it always holds that $oc < oc_{max}$, i.e. the current optimization cycle which was started by the predecessor has still to be completed in the form of an additional initiation.

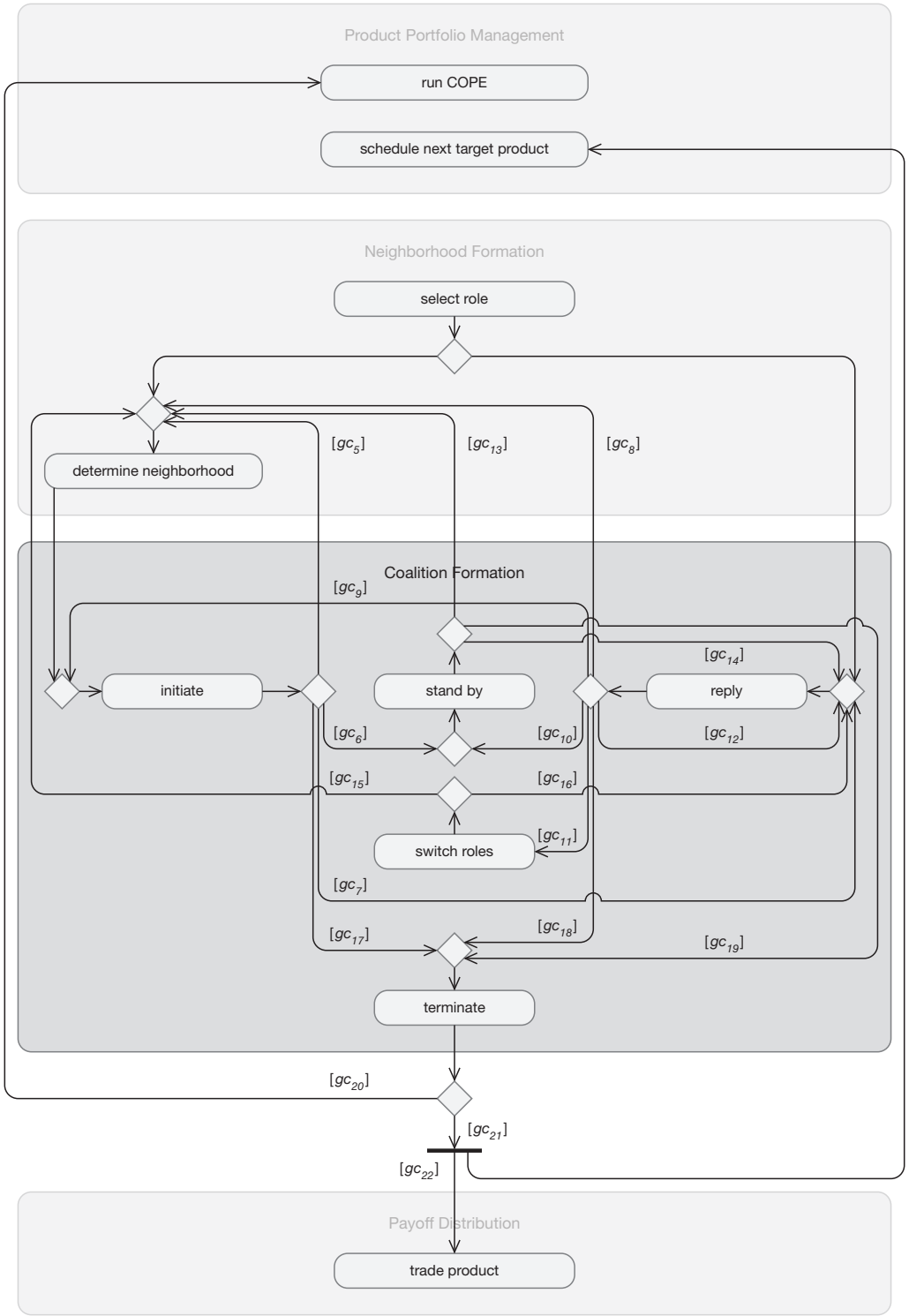


Figure 4.15: Activity of coalition formation.

ID	condition[a]
gc_5	$(\neg exp(t_{f,p_{tmp}}) \wedge newRep(C_{RI}))$
$gc_{6,1}$	$(\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_I) \wedge suc(C_I) \wedge oc = oc_{max})$
$gc_{6,2}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge newRep(C_{RR}) \wedge suc(C_{RR}) \wedge oc = oc_{max})$
$gc_{6,3}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge newNonRep(C_R))$
$gc_{7,1}$	$(\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_I) \wedge \neg suc(C_I))$
$gc_{7,2}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_I) \wedge suc(C_I) \wedge oc < oc_{max})$
$gc_{7,3}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge newRep(C_{RR}) \wedge \neg suc(C_{RR}))$
$gc_{7,4}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge newRep(C_{RR}) \wedge suc(C_{RR}) \wedge oc < oc_{max})$
$gc_{8,1}$	$(\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_{RI}) \wedge \neg suc(C_{RI}) \wedge i_u = i_{u,max} \wedge ext(N))$
$gc_{8,2}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_{RI}) \wedge suc(C_{RI}) \wedge i_u = i_{u,max})$
$gc_{8,3}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge newRep(C_I) \wedge \neg suc(C_I))$
$gc_{8,4}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge newRep(C_I) \wedge suc(C_I) \wedge oc < oc_{max})$
gc_9	$\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_{RI}) \wedge i_u < i_{u,max}$
$gc_{10,1}$	$(\neg exp(t_{f,p_{tmp}}) \wedge newRep(C_I) \wedge suc(C_I) \wedge oc = oc_{max})$
$gc_{10,2}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge newNonRep(C_I))$
$gc_{10,3}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_{RR}) \wedge suc(C_{RR}) \wedge oc = oc_{max})$
$gc_{11,1}$	$(\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_{RI}) \wedge \neg suc(C_{RI}) \wedge i_u = i_{u,max} \wedge \neg ext(N))$
$gc_{11,2}$	$\vee(\neg exp(t_{f,p_{tmp}}) \wedge rep(C_{RR}) \wedge exp(t_{bi}) \wedge bi < bi_{max} \wedge rand(0,1) < P(bi))$
$gc_{12,1}$	$\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_{RR}) \wedge \neg suc(C_{RR})$
$gc_{12,2}$	$\neg exp(t_{f,p_{tmp}}) \wedge stillRep(C_{RR}) \wedge suc(C_{RR} \wedge oc < oc_{max})$
gc_{13}	$\neg exp(t_{f,p_{tmp}}) \wedge initiator()$
gc_{14}	$\neg exp(t_{f,p_{tmp}}) \wedge \neg initiator()$
gc_{15}	$\neg exp(t_{f,p_{tmp}}) \wedge initiator()$
gc_{16}	$\neg exp(t_{f,p_{tmp}}) \wedge \neg initiator()$
gc_{17}	$exp(t_{f,p_{tmp}})$
gc_{18}	$exp(t_{f,p_{tmp}})$
gc_{19}	$exp(t_{f,p_{tmp}})$
gc_{20}	$\neg suc(C) \wedge p_{tmp} \notin P_{tmp,base}$
$gc_{21,1}$	$(\neg suc(C) \wedge p_{tmp} \in P_{tmp,base})$
$gc_{21,2}$	$\vee suc(C)$
gc_{22}	$suc(C)$

[a] $exp(t)$ returns true if time interval t has expired, false otherwise.
$ext(N)$ returns true if neighborhood N is extendable, false otherwise.
$newNonRep(C)$ returns true if after a regrouping a is new non-representative of C, false otherwise.
$newRep(C)$ returns true if after a regrouping a is new representative of C, false otherwise.
$rand(0, 1)$ returns a uniformly distributed random number between 0 and 1.
$rep(C)$ returns true if a is representative of C, false otherwise.
$stillRep(C)$ returns true if after a regrouping a is still representative of C, false otherwise.
$suc(C)$ returns true if coalition C is successful, false otherwise.

Table 4.1: Guard conditions for activity coalition formation.

Contrary, if the agent took on the role of a responder and its new coalition C_{RR} is not successful yet, it proceeds by replying to a call for proposal ($gc_{7,3}$). The same holds if the target product is already fulfilled but the maximum number of optimization cycles has not been reached through a corresponding number of additional replies ($gc_{7,4}$). However, if the adopted coalition is successful and all optimization efforts have been completed, the agent switches into standby mode rejecting all further regrouping requests ($gc_{6,2}$).
When the initiator continues the process as representative of its own coalition and replies to a call for proposal ($gc_{7,1}$ or $gc_{7,2}$), it first validates the deadlines of the received messages and verifies that the members of the transmitted coalitions have not already joined the own one in the course of previous regrouping activities. This situation may arise because initiators only respond to a single call before proceeding with the next action which might result in a temporal gap between the receival and the evaluation of a message throughout which the sender can join another coalition which then transfers the agent to the initiator in the course of a further regrouping. However, such situations are typically very rare as these activities have to be completed before the deadline of a sent CFP has expired. When evaluating the received messages, the initiator replies to the first valid call through application of interaction protocol 4.12.
Given that the current formation interval has still not expired, the following action then again depends on the outcome of the negotiations. If the initiator stays representative of its own coalition C_{RI} and the number of consecutive unsuccessful initiations i_u within its current neighborhood N has not reached a specified maximum $i_{u,max}$, it proceeds by starting another negotiation (gc_9). The start of the new negotiation is independent from the success of C_{RI} because even in case of a successful product fulfillment the current optimization cycle has still to be completed. An initiation is generally considered unsuccessful if it does not result in a value optimizing regrouping. This is either the case if no neighbors are left after updating the neighborhood at the beginning of interaction protocol 4.12 or no proposal is received as reply to the following call. Setting parameter $i_{u,max}$ to a high value generally provides the potential to limit the final scope of a coalition by preventing a premature neighborhood extension and increasing the chance of a cooperation between nearer neighbors.
In contrast, if the maximum number of unsuccessful initiations has been reached, the success of the coalition is again relevant for the choice of the initiator's next action. More precisely, if the maximum number has been reached and coalition C_{RI} is still unsuccessful but the current neighborhood N is still extendable, the initiator expands the latter by a specified extension count ext_N and starts another negotiation given the new size ($gc_{8,1}$). However, if N already covers the whole grid ($gc_{11,1}$), the agent finally changes its strategy by switching its interaction role and waiting for appropriate calls as a responder (gc_{16}). In contrast, if the maximum number of unsuccessful initiations has been reached and coalition C_{RI} is already successful, the extendability of the neighborhood is considered irrelevant and the agent completes the current optimization cycle by simply starting another negotiation ($gc_{8,2}$).
If after the reply the initiator does not stay representative of its own coalition, it either becomes representative or non-representative of the initiating coalition C_I. In the former

case, it restarts the activity if C_I is unsuccessful or if it is successful but has not conducted the maximum number of optimization cycles yet ($gc_{8,3}$ and $gc_{8,4}$). However, if the target product is already fulfilled and all optimization efforts have been completed, the agent considers the formation process as finished and switches into standby mode ($gc_{10,1}$). The same holds if it leaves the negotiations as non-representative member of C_I, in which case it is still subject to further regroupings, however ($gc_{10,2}$).
In contrast to an initiator, a responder starts the activity of coalition formation with a reply to a call for proposal through application of interaction protocol 4.12. If formation interval $t_{f,p_{tmp}}$ has not yet expired and the agent leaves the negotiations as new representative or non-representative of the initiating coalition C_I, its further behavior is equal to that of a switching replying initiator as described above ($gc_{8,3}$, $gc_{8,4}$, $gc_{10,1}$, and $gc_{10,2}$). Contrary, if it stays representative of its own coalition C_{RR} and the latter is either unsuccessful or successful but not all additional optimization efforts have been completed yet, it proceeds by replying to the next CFP which deadline has not yet expired ($gc_{12,1}$ and $gc_{12,2}$). Correspondingly, if the coalition fulfills its target product and the additional optimization efforts have been completed, the agent considers its goal as achieved and switches into standby mode ($gc_{10,3}$). However, if the target product is not fulfilled and the agent has not received a call for a specified amount of time t_{bi}, it finally switches its interaction role with a defined probability $P(bi)$ for a maximum number of bi_{max} times ($gc_{11,2}$) in order to proactively participate in the negotiations and prevent an unsuccessful outcome caused by sole passivity (gc_{15}). Associating the role change with a specific probability allows to control the global rate of initiators r_{init} throughout the whole formation interval t_{CF} by means of a regulatory hard constraint hc_a.
When the formation interval for the considered product template p_{tmp} has expired and the agents have finished the current negotiations (gc_{17}, gc_{18}) or are standing by (gc_{19}), they finally terminate the activity of coalition formation. Depending on the success of their coalition, they then reenter the activity of product portfolio management with or without the goal to redefine their product portfolio PP through a rerun of the COPE algorithm. In this context, recall from Section 4.1.2 that the input parameters of the latter include template catalog P_{tmp} comprising all product templates which an agent generally considers for product specification as well as base catalog $P_{tmp,base}$ including those templates of P_{tmp} which product horizons cover the smallest tradable interval and thus guarantee the creation of a consistent template portfolio. Throughout execution, both sets are used by the algorithm in order to create a reduced template catalog $P_{tmp\backslash base} = P_{tmp} \backslash P_{tmp,base}$ comprising all templates which it considers for optimizing the current template portfolio through template replacement. As already illustrated by means of the initially specified use case, the base catalog of the EPEX SPOT template catalog as shown in Figure 4.2 is for instance given by the 24 single hour products, whereas the reduced template catalog is given by the remaining block products.
With regard to these parameters, a redefinition of a product portfolio is generally only possible if the currently considered product template p_{tmp} is not part of base catalog $P_{tmp,base}$ as in this case it is not essential for the specification of a consistent template portfolio and can be safely removed from template catalog P_{tmp}. Thus, if a formation

process did not lead to a successful fulfillment of its target product and it holds that $p_{tmp} \notin P_{tmp,base}$, an agent updates its template catalog accordingly and conducts the activity of product portfolio management again by rerunning the COPE algorithm and scheduling the next target product (gc_{20}). To illustrate, assume for instance that the producer from our exemplary use case strived for the provision of a peakload product but finally failed in the attempt to form a successful coalition. However, as the related template is not part of the base catalog, the agent is able to remove it from the template catalog and run the COPE algorithm again in order to generate a new product portfolio without the peakload product. As this approach is not applicable if the considered template is part of the base catalog, in this case an agent accepts the result and proceeds with the overall process by scheduling its next target product immediately ($gc_{21,1}$). The same holds if it finishes the process as a member of a successful coalition. However, as in this case the agent is committed to the trade of the fulfilled product p, it first removes all operation schedules os_U from its schedule space OSS_U which do not match the values of its contribution $con_{a,p}$ and replaces those templates of its base catalog with the considered template p_{tmp} which are temporally covered by the latter. Moreover, it updates $P_{tmp\backslash base}$ by removing both p_{tmp} and all temporally overlapping templates. These measures guarantee on the one hand that the agent is finally able to contribute to the product as promised and on the other hand that the latter is actually part of its product portfolio and not replaced by COPE in the course of further redefinitions within the current formation period t_{CF}. To illustrate, assume for instance that our producer formed a successful coalition for a product $p_{Middle-Night}$ (cf. Figure 4.2). When terminating the formation process, it thus updates its operation schedule space by removing all schedules which are not in line with its contribution $con_{a,p_{Middle-Night}}$. Moreover, it updates its base catalog by replacing the first three single hour templates by $p_{tmp,Middle-Night}$ and then removes the latter from $P_{tmp\backslash base}$ along with the temporally overlapping templates $p_{tmp,Night}$, $p_{tmp,Off-Peak1}$, $p_{tmp,Off-Peak}$, and $p_{tmp,Baseload}$. As $p_{tmp,Middle-Night}$ is now part of the base catalog and can not be replaced by any other template, it is guaranteed that the fulfilled target product is part of the product portfolio even in case of a further redefinition.

The product p which can be traded by a coalition if a formation process was successful is generally defined by its cumulative contribution con_C, i.e. $\forall t_{pr}^{(i)} \in T_{pr}^{(p)} : (e_p^{(i)}, err_p^{(i)}, c_p^{(i)}) = (e_C^{(i)}, err_C^{(i)}, c_C^{(i)})$. In the worst case, the values of p can thus deviate from the ones of the originally pursued target product p_{tar} to the extent as defined by the tolerance bands $par_{tar,tol}$. In order to perform all actions associated with the cooperative trade of the product, an agent finally performs the last activity of DYCE which is conducted in parallel to product portfolio management (gc_{22}).

4.4 Payoff Distribution

Having successfully formed a coalition, an agent finally conducts the last activity of the overall process in order to perform all actions which are associated with the trade of the fulfilled product as well as the division of the resulting payoff. In the following

sections, we first specify the considered problem and discuss the related computational complexity in view of the game-theoretical concepts applied for its solution. Afterwards, we describe the activity of payoff distribution in detail and provide a distribution model which is based on concepts from the domain of cooperative game theory and allows for a fair division of a coalition's utility according to the individual contributions of its members.

4.4.1 Problem Specification and Computational Complexity

If the interactions of the previous activity led to the formation of a successful coalition C, the members finally strive for a trade of the fulfilled product p. In case the trade was successful, the coalition then receives or pays a total cost constituting its final payoff $\check{\rho}$. Assuming that the product horizon of p is given by $T_{pr}^{(p)} = \{t_{pr}^{(0)}, \ldots, t_{pr}^{(j_{max})}\}$, the latter is generally defined as

$$\check{\rho} = \sum_{i=0}^{j_{max}} e_p^{(i)} \cdot \check{c}^{(i)}, \tag{4.41}$$

where $\check{c}^{(i)}$ is the final contractually specified cost per unit electrical energy in product interval $t_{pr}^{(i)}$. For instance, if a coalition sells the quantity of $e_p^{(i)} = 1000\,\text{kWh}$ at a price of $c^{(i)} = 0.1\,€/\text{kWh}$, its final payoff amounts to $\check{\rho} = 100\,€$. Generally, $\check{\rho}$ does not necessarily equal the original payoff which was initially specified by the coalition for the product which is given by

$$\rho = \sum_{i=0}^{j_{max}} e_p^{(i)} \cdot c_p^{(i)}. \tag{4.42}$$

For instance, if p is placed as a bid at an exchange, the market clearing price $\check{c}^{(i)}$ typically deviates from the original cost $c_p^{(i)}$.
The payoff finally determines the utility which a coalition gains from a trade. For a coalition of producers C_p and a coalition of consumers C_c this is defined as

$$utility_C(p) = \sum_{i=0}^{j_{max}} utility_C^{(i)}(p) = \begin{cases} \sum_{i=0}^{j_{max}} e_p^{(i)} \cdot \check{c}^{(i)} - e_p^{(i)} \cdot c_p^{(i)} = \check{\rho} - \rho & \text{if } C \text{ is } C_p, \\ \sum_{i=0}^{j_{max}} e_p^{(i)} \cdot c_p^{(i)} - e_p^{(i)} \cdot \check{c}^{(i)} = \rho - \check{\rho} & \text{if } C \text{ is } C_c. \end{cases} \tag{4.43}$$

I.e., the utility of a coalition is defined as sum of interval-related utilities $utility_C^{(i)}(p)$, where for a producer coalition the latter are calculated by subtracting the original from the final payoffs and for a consumer coalition by subtracting the final from the original payoffs. These definitions are consistent with Equation 4.1 which was used by an agent prior to the formation process in order to estimate the expected utility of a given product template p_{tmp} and operation schedule os_U based on a price prediction $c_{p_{tmp}}^{(i)}$. In the

following, we generally assume that $utility_C^{(i)}(p) \geq 0$, i.e. we suppose that a producer coalition only sells at a price which equals or exceeds the originally specified cost $c_p^{(i)}$ and a consumer coalition only buys at a price which is less than or equal to the originally specified cost. If the product is traded at an exchange which matches block products on an all or none basis (like the EPEX SPOT), this assumption is inherently fulfilled by the clearing mechanism itself.
As main action of payoff distribution, a successful coalition $C = \{a_1, \ldots, a_n\}$ finally divides the payoff $\check{\rho}$ into individual shares $(x_1, \ldots, x_n)$ relating to its members (cf. Definition 3.16). The complexity of the action is generally determined by the requirements put on the resulting division. In this regard, we generally demand that the utility of a coalition is distributed in a fair way (cf. Chapter 1.3). In the following section, we thus propose an approach which allows to realize this goal based on concepts from cooperative game theory. More precisely, we define a conjunction of weighted voting games reflecting the members' contributions to their coalition and then apply the Shapley value in order to provide a fair division of the gained utility. However, calculating the Shapley value for a single weighted voting game is generally #P-hard [DP94]. Thus, the described model allows the application of an approximation method [FWJ08] in order to enable an efficient assessment of distributions even in case of large coalition sizes.

4.4.2 Activity Description

The activity of payoff distribution and its relation to the other DYCE activities is depicted by the UML diagram in Figure 4.16. As shown, an agent starts the process by performing all tasks which are associated with the trade of the product for which a successful coalition was formed. In this context, all contractual issues are coordinated by the representative of the coalition which first requests binding commitments from all members with regard to their individual contributions as specified in the course of coalition formation. Having received the commitments, it then conducts the actual trade by transmitting a corresponding bid to a bilateral trading partner or a central trading platform like a pool or an exchange. The transfer has to be finished until a specified time of submission t_{sub} (e.g. the time of order book closure of an exchange) after which the recipient either accepts or declines the made offer. In the former case, both trading partners complete the trade by conducting all required business processes including the related transactions. Afterwards, the representative sends a success message to all members of the coalition which then schedule their units according to the committed contributions.
If a trade was successful, a coalition physically fulfills its product in the specified product horizon $T_{pr}^{(p)}$ as second action of the activity. I.e., a producer coalition jointly feeds in the specified amount of electrical energy, while a consumer coalition takes it from the grid. After physical fulfillment, the representative requests the related measurements from the responsible measurement service provider in order to determine the final electricity amounts $\check{e}_U^{(i)}$ which were generated or consumed by each member in the product intervals $t_{pr}^{(i)} \in T_{pr}^{(p)}$. These potentially deviate from the scheduled values $e_U^{(i)}$ because of unforseen

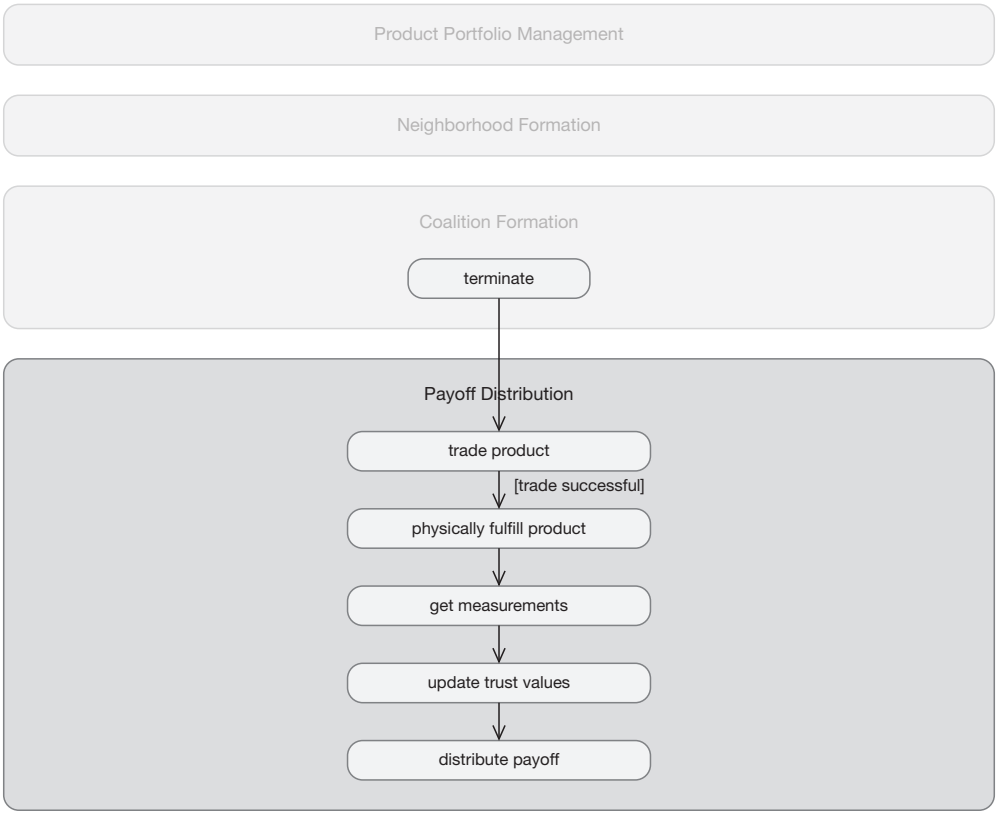


Figure 4.16: Activity of payoff distribution.

events which occurred in the course of fulfillment. The resulting final errors are defined as

$$\check{err}_U^{(i)} = |e_U^{(i)} - \check{e}_U^{(i)}|. \tag{4.44}$$

These may differ from the originally specified values $err_U^{(i)}$ and thus represent a measure of the actual reliability which was provided by an agent in the course of physical fulfillment. As this criterion influences an agent's trustworthiness (cf. Section 4.2.2), the representative calculates the errors of the whole coalition based on the queried measurements and informs the other members about the results. As next step of the activity, the agents then update their maintained trust values taking into account the received information. Note that this approach incentivizes participants to declare their electricity amounts $e_U^{(i)}$ both correctly (in terms of an accurate forecast) and truthfully because it increases their chance to find cooperation partners in the course of future formation processes. This, in turn, promotes the reliability of coalitions as it minimizes deviations from contractually specified quantities throughout physical fulfillment.

As final action of the activity, the payoff $\check{\rho}$ which resulted from the previous trade is distributed among the members. As the representative is in charge of all contractual issues, it is also responsible for calculating the division and initiating the required business processes. To this end, it applies a dedicated distribution model in order to determine the distribution vector $\mathbf{x}_C = (x_1, \ldots, x_n)$ specifying the individual shares of all members. As specified in Section 1.3, we generally expect that this division is fair and provides an incentive for agents to declare their contributions correctly and truthfully in the first place. In this regard, Chalkiadakis et al. [CRK+11] propose a payment mechanism for Virtual Power Plants which divides a received financial amount among participating DERs. While the approach indeed incentivizes members to provide accurate predictions of their expected production, it does not allow for a fair division in terms of a mathematically justified notion. Moreover, in order to promote economic efficiency, it would be desirable to award units based their generation and consumption cost as well. In the following, we thus provide a distribution model which satisfies all of our specified requirements. As general approach, we determine distribution vector $\mathbf{x}_C$ by assigning to each agent its total schedule cost plus a fair percentage of the coalition's utility according to the benefit of the agent's final contribution $\check{con}_{a,p} = (\check{e}_U^{(i)}, \check{err}_U^{(i)}, c_U^{(i)})$. I.e., producers receive their specified cost plus a fair fraction of the gained surplus, while consumers pay their specified price minus a fair fraction. Because the fair fractions are determined based on the final electricity amounts $\check{e}_U^{(i)}$ and errors $\check{err}_U^{(i)}$, agents are incentivized to declare their produced or consumed amounts correctly and truthfully which again results in more reliable coalitions. More precisely, assuming that the product horizon of the traded product p is given by $T_{pr}^{(p)} = \{t_{pr}^{(0)}, \ldots, t_{pr}^{(j_{max})}\}$, the individual share of a coalition member a_k is calculated as

$$x_k = \sum_{i=0}^{j_{max}} x_k^{(i)} = \sum_{i=0}^{j_{max}} c_U^{(i)} \cdot e_U^{(i)} + \phi_k^{(i)}(C, v_g) \cdot utility_C^{(i)}(p), \tag{4.45}$$

with $\phi_k^{(i)}(C, v_g)$ being a fair percentage in terms of the Shapley value as described in more detail below. The overall share of an agent is thus defined as sum of interval-related shares $x_k^{(i)}$, where an interval-related share is the sum of the corresponding schedule cost $c_U^{(i)} \cdot e_U^{(i)}$ (as specified by contribution $con_{a,p}(t_{pr}^{(i)})$) and a fair fraction of the coalition's utility $\phi_k^{(i)}(C, v_g) \cdot utility_C^{(i)}(p)$.

In order to determine the fair percentage $\phi_k^{(i)}(C, v_g)$, the model integrates concepts from the domain of cooperative game theory. As a branch of game theory, the field studies settings in which self-interested players are able to make binding agreements (like legally valid contracts) which allow them to rationally cooperate and act as a group [SLB09, OR94, Woo09]. With regard to DYCE, the requirement of binding agreements is generally fulfilled because members make final commitments to a coalition as described above. A main question which is addressed by cooperative game theory is how to distribute the utility resulting from a cooperative action among the players. A model which allows for this kind of analysis is given by a *coalitional game with transferable utility* (I, v_g), where

- $I = \{a_1, \ldots, a_n\}$ is a set of *players* (or agents), and
- $v_g : 2^I \rightarrow \mathbb{R}$ is a function (also referred to as *characteristic function*) assigning a freely dividable utility $v_g(S)$ to each nonempty coalition $S \subseteq I$[15].

A well-known and widely studied solution concept applicable to this type of game is the Shapley value which was proposed by Lloyd S. Shapley in 1953 [Sha88]. Being devised as a function, it assigns a fair distribution vector to a given coalitional game (I, v_g), where fairness is defined in terms of specific axioms representing properties of the resulting division. Before we discuss the axiomatic characterization of the Shapley value in detail, we first provide its formal definition. To this end, let the *marginal contribution* of an agent a_k to a coalition $S \subset I$, with $a_k \notin S$, be defined as

$$\Delta_k(S) = v(S \cup \{a_k\}) - v(S). \tag{4.46}$$

I.e., $\Delta_k(S)$ captures the utility which an agent adds to a coalition through its participation. Given this, the Shapley value of an agent a_k is defined as

$$\phi_k(I, v_g) = \frac{1}{|I|!} \sum_{S \subseteq I \setminus a_k} |S|! \cdot (|I| - |S| - 1)! \cdot \Delta_k(S). \tag{4.47}$$

Generally, the above formula determines the average marginal contribution of an agent to a coalition I calculated over all possible sequences in which the members can join, where all sequences are equally likely. This is achieved by calculating for each possible subset $S \subseteq I \setminus a_k$ the agent's marginal contribution when it joins as member number $|S| + 1$ multiplied by the $|S|!$ possibilities its predecessors could have joined and the $(|I| - |S| - 1)!$ possibilities its successors can join afterwards. Adding up the results and dividing the sum by the total number of all sequences $|I|!$ finally yields the average marginal contribution of the agent.
I.e., the distribution of a coalitional game (I, v_g) is calculated according to the utility which each agent contributes to I on average. The underlying notion of fairness which specifies related properties of the division is now defined as follows. First, let an agent a_k be termed a *null player* if it does not contribute a utility to any coalition, i.e. if $v_g(\{S \cup a_k\}) = v_g(S)$ for all coalitions $S \subset I$ with $a_k \notin S$. Moreover, let two agents a_k and a_l be termed *interchangeable* if they contribute the same utility to any coalition, i.e. if $v_g(S \cup \{a_k\}) = v_g(S \cup \{a_l\})$ for all coalitions $S \subset I$ with $a_k \notin S \wedge a_l \notin S$. The Shapley value is then considered fair as it satisfies the following axioms:

Efficiency The utility of a coalition is completely distributed among the players, i.e. $\sum_{k=1}^{|I|} \phi_k(I, v_g) = v_g(I)$.

Null player If a player a_k is a null player, then its share is zero, i.e. $\phi_k(I, v_g) = 0$.

[15] In order to avoid confusion, we adapt some common symbols to the notation defined in Chapter 3. For instance, while the characteristic function is typically denoted by v, we use the subscripted symbol v_g in order to distinguish it from function v introduced in Definition 3.14.

Symmetry If two players a_k and a_l are interchangeable, then their shares are equal, i.e. $\phi_k(I, v_g) = \phi_l(I, v_g)$.

Additivity The shares of any player $a_k \in I$ in two separate games $(I, v_{g,1})$ and $(I, v_{g,2})$ relate to each other such that $\phi_k(I, v_{g,1} + v_{g,2}) = \phi_k(I, v_{g,1}) + \phi_k(I, v_{g,2})$, where $(I, v_{g,1} + v_{g,2})$ is a combined game which characteristic function is given by $(v_{g,1} + v_{g,2})(S) = v_{g,1}(S) + v_{g,2}(S)$ for every coalition $S \subseteq I$.

Besides satisfying the above axioms, the Shapley value is also the only distribution scheme possessing these properties. Thus, it allows to determine both a fair and unique division of a coalition's utility.
Given this solution concept fulfilling our initially specified requirement of fairness, the remaining question to answer is how to model the underlying coalitional game (I, v_g) so that it fits our considered context. As our intention is to distribute the utility of a coalition C according to the members' individual contributions to a given product p, a first idea could be to define a coalitional game (C, v) for each product interval $t_{pr}^{(i)} \in T_{pr}^{(p)}$, where v represents the initially defined evaluation function as specified in Equation 4.36. As the latter measures the degree of product fulfillment and a coalition might not have matched its pursued target values $e_{p_{tar}}^{(i)}$, $err_{p_{tar}}^{(i)}$, and $c_{p_{tar}}^{(i)}$ exactly, this approach would require to set these to the cumulative values $e_C^{(i)}$, $err_C^{(i)}$, and $c_C^{(i)}$ before calculating the distribution in order to ensure that the value of C becomes maximal (i.e. $v(C) = 1$). Because of the efficiency axiom, an application of the Shapley value to the resulting games would then yield interval-related distribution vectors which shares $\phi_k^{(i)}(C, v_g)$ would sum up to one (as $v(C) = 1$) and thus represent fair percentages as required by Equation 4.45. However, as the calculation of Equation 4.47 is highly complex, this model would only be applicable if the size of a coalition is very small. Moreover, it would not be flexibly extendable by other criteria which are to be taken into account for value distribution as well.
In order to tackle these problems, we thus take an alternative approach and define a certain type of coalitional game for which the Shapley value can be efficiently approximated using the algorithm described in [FWJ08] and that is flexibly extendable by further criteria if desired. Before we describe the specific model in detail, we start by introducing the required game-theoretical concepts [EGGW09]. First, let $(I; \mathbf{w}; q)$ denote a *weighted voting game*, where $I = \{a_1, \ldots, a_n\}$ is a set of players, $\mathbf{w} = (w_1, \ldots, w_n) \in \mathbb{R}_{\geq 0}^n$ is a vector of *weights* assigned to the players, and $q \in \mathbb{R}_{\geq 0}$ is a *quota*. The game's characteristic function is defined as $v_g : 2^I \to \{0, 1\}$, with

$$v_g(S) = \begin{cases} 1 & \text{if } \sum_{i=1}^{|S|} w_i \geq q, \\ 0 & \text{otherwise.} \end{cases} \tag{4.48}$$

A coalition S is said to be *winning* if the summed weights are equal to or greater than q (i.e., if $v_g(S) = 1$), and said to be *losing* otherwise. Moreover, we term a player a_i a *swing player* for a coalition S if S is losing but $S \cup a_i$ is winning. Weighted voting games are typically used to analyze political decision processes in which a group of

decision makers needs a minimum number of votes in order to take a certain action like passing a law. However, as a weight can represent an arbitrary resource, they also allow to study alternative settings in which a coalition is able to achieve a specific goal if its joint quantity of that resource meets a given threshold.

An extended form of a weighted voting game is given by a *k-vector weighted voting game* $(I, \mathbf{w}^1, \ldots, \mathbf{w}^k, \mathbf{q})$ (also referred to as *weighted k-majority game*) which is defined by a set of players $I = \{a_1, \ldots, a_n\}$, a vector of vectors $(\mathbf{w}^1, \ldots, \mathbf{w}^k)$ specifying lists of weights $\mathbf{w}^j = (w_1^j, \ldots, w_n^j)$, and a vector of quotas $\mathbf{q} = (q^1, \ldots, q^k)$. Thus, a k-vector weighted voting game implicitly defines a set of weighted voting games $(I; \mathbf{w}^j; q^j)$, $1 \leq j \leq k$. The game's characteristic function is defined as $v_g : 2^I \to \{0, 1\}$, where

$$v_g(S) = \begin{cases} 1 & \text{if } \sum_{i=1}^{|S|} w_i^j \geq q^j, 1 \leq j \leq k, \\ 0 & \text{otherwise.} \end{cases} \tag{4.49}$$

I.e., in a k-vector weighted voting game a coalition S is winning if it wins in all comprised weighted voting games. This type of coalitional game thus allows to analyze situations in which a set of players is able to achieve a specific goal if it fulfills all of a set of required conditions.

Given these definitions, the basic idea with regard to our considered problem is now to create a k-vector weighted voting game for each product interval $t_{pr}^{(i)} \in T_{pr}^{(p)}$ and define the weights in such a way that they represent the contributions of the members to the physically fulfilled product p. Calculating the Shapley values of the games then yields the desired fair percentages $\phi_k^{(i)}(C, v_g)$ which reflect the relevance of the contributions.

More precisely, for a given product interval $t_{pr}^{(i)}$, we specify a 3-vector weighted voting game $(C, \mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3, \mathbf{q})$ which implicitly defined games $(C; \mathbf{w}^j; q^j)$ address the product attributes electricity amount, error, and cost. As already mentioned above, we generally use the final electricity amounts $\check{e}_U^{(i)}$ and errors $e\check{r}r_U^{(i)}$ for the model in order to award the agents according to their actual behavior. The first weighted voting game $(C; \mathbf{w}^1; q^1)$ thus refers to the electricity amounts $\check{e}_U^{(i)}$ which were produced or consumed by the members of the coalition. The individual weights w_i^1 are determined by standardizing the absolute values $|\check{e}_U^{(i)}|$ using the linear function shown on the left hand side of Figure 4.17 which is formally defined as $\lambda^+ : \mathbb{R}_{\geq} \to \{0, 1\}$, with

$$\lambda^+(x) = \begin{cases} \frac{1}{|x_{max}|} \cdot |x| & \text{if } 0 \leq x \leq x_{max}, \\ 0 & \text{otherwise,} \end{cases} \tag{4.50}$$

where for the first game it holds that $x_{max} = \sum_{U \in U_C} |\check{e}_U^{(i)}|$. Thus, λ^+ assigns higher weights to those agents which made more valuable contributions to the coalition in order to allow for higher final shares as described in more detail below. Generally, the quota of the game can be chosen from the interval $]0, x_{max}]$ which endpoints guarantee that the Shapely value can be reasonably applied. In particular, setting the upper bound to x_{max} ensures that the whole coalition wins (i.e. $v_g(C) = 1$). As the quota influences the final distribution, it has to be agreed upon by all coalition members using an appropriate

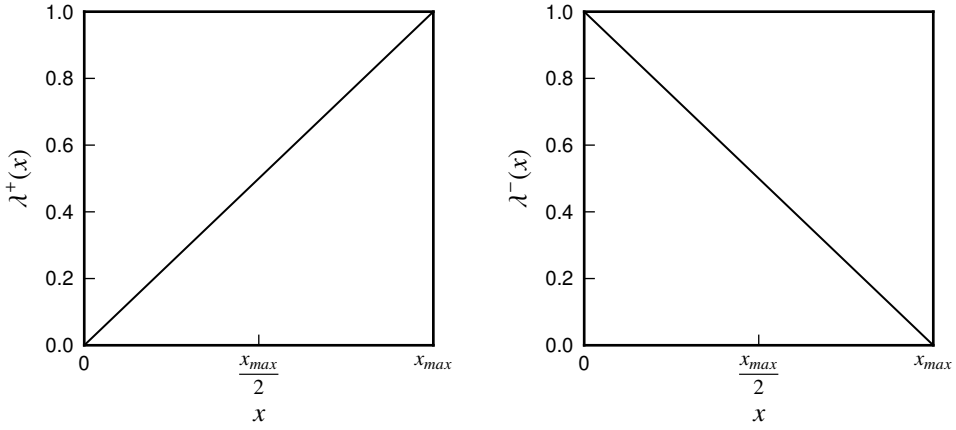


Figure 4.17: Standardization functions λ^+ (left) and λ^- (right).

mechanism (like negotiation or voting) or prescribed on global system level by means of a regulatory hard constraint hc_a. To illustrate its impact, suppose for instance a coalition $C = \{a_1, a_2, a_3, a_4\}$ which members fed-in the amounts of $\check{e}^{(i)}_{U_1} = 10\,\text{kWh}$, $\check{e}^{(i)}_{U_2} = 20\,\text{kWh}$, $\check{e}^{(i)}_{U_3} = 30\,\text{kWh}$, and $\check{e}^{(i)}_{U_4} = 40\,\text{kWh}$. As in this case $x_{max} = 100\,\text{kWh}$, these are mapped by function λ^+ to the weights $w_1 = 0.1$, $w_2 = 0.2$, $w_3 = 0.3$, and $w_4 = 0.4$, respectively. The Shapley values for this setting and different quotas q are listed besides two other examples in Table 4.2. As shown, the quota influences the final shares $\phi_k(N, v_g)$, where there is no clear pattern associated with its increase (like a more balanced distribution). Moreover, it can be seen that a higher weight w_i does not necessarily lead to a higher share. I.e., the latter generally depends on both the variance of the weights and the quota which thus has to be chosen with care taking into account the given economic and technological conditions like the design of the market or the specific types of the members' units.

The second weighted voting game $(C; \mathbf{w}^2; q^2)$ addresses the errors which occurred in the course of physical fulfillment. As we generally strive for a minimization of deviations and want to award the agents accordingly, the weights w^2_i are determined by transforming the final errors $\check{err}^{(i)}_U$ into standardized values using an alternative linear function which reverses its input as shown on the right hand side of Figure 4.17. Formally, it is defined as $\lambda^- : \mathbb{R} \to \{0, 1\}$, with

$$\lambda^-(x) = \begin{cases} -\frac{1}{x_{max}} \cdot x - \frac{1}{x_{max}} \cdot x_{max} & \text{if } 0 \leq x \leq x_{max}, \\ 0 & \text{otherwise,} \end{cases} \tag{4.51}$$

where for the second game it holds that $x_{max} = \sum_{U \in U_C} \check{err}^{(i)}_U$. As λ^- maps smaller errors to higher weights and vice versa, it allows to assign higher rewards to those agents which

w_1	w_2	w_3	w_4	q	$\phi_1(N, v_g)$	$\phi_2(N, v_g)$	$\phi_3(N, v_g)$	$\phi_4(N, v_g)$
0.1	0.2	0.3	0.4	0.2	0	0.333	0.333	0.333
				0.4	0.083	0.083	0.25	0.583
				0.6	0.083	0.25	0.25	0.417
				0.8	0.083	0.083	0.417	0.417
				1.0	0.25	0.25	0.25	0.25
0.1	0.1	0.4	0.4	0.2	0.83	0.83	0.417	0.417
				0.4	0	0	0.5	0.5
				0.6	0.167	0.167	0.333	0.333
				0.8	0	0	0.5	0.5
				1.0	0.25	0.25	0.25	0.25
0.1	0.1	0.1	0.4	0.2	0.167	0.167	0.167	0.5
				0.4	0	0	0	1
				0.6	0	0	0	1
				0.8	0.083	0.083	0.83	0.75
				1.0	0.25	0.25	0.25	0.25

Table 4.2: Exemplary Shapley values for different games.

provided a higher reliability in the course of physical fulfillment in the sense of a lower deviation from the initially specified electricity amount. Again, the quota of the game can be chosen from the interval $]0, x_{max}]$.
Finally, the last weighted voting game $(C; \mathbf{w}^3; q^3)$ addresses the cost values which were specified by the members for their provided or consumed electricity amounts. As in this regard producers and consumers have contrary goals, the weights w_i^3 are determined by using different standardization functions. More precisely, if the given coalition is a producer coalition C_p and thus strives for a minimization of its cost, we apply function λ^- in order to allow for higher rewards for those agents with lower generation cost $c_U^{(i)}$. Contrary, if the coalition is a consumer coalition C_c and takes advantage of agents which are willing to pay higher prices, we determine the weights by using function λ^+. For both types of coalitions it holds that $x_{max} = \sum_{U \in U_C} \check{c}_U^{(i)}$, while the quota q^3 can be chosen from the interval $]0, x_{max}]$. The settings of all three games are summed up in Table 4.3.
Given the resulting 3-vector weighted voting game $(C, \mathbf{w}^1, \mathbf{w}^2, \mathbf{w}^3, \mathbf{q})$[16] for product interval $t_{pr}^{(i)}$, we are finally able to determine the related Shapley values $\phi_k^{(i)}(C, v_g)$, where each single share reflects the importance of the corresponding agent in terms of its potential to enable the coalition to meet the defined quotas. More precisely, recall from Equation 4.47 that the calculation of the Shapley value of an agent a_k involves the assessment of its marginal contribution $\Delta_k(S)$ to each subset $S \subseteq C \setminus a_k$. Given our specified game,

[16] Note that this model is flexibly extendable. For instance, if the time of an agent's joining is to be taken into account as distribution criteria, it can be modeled as own weighted voting game and integrated into the combined game.

	w_i			
game	C_p	C_c	x_{max}	q
$(C; \mathbf{w}^1; q^1)$	$\lambda^+(x)$	$\lambda^+(x)$	$\sum_{U \in U_C} \check{e}_U^{(i)}$	$]0, x_{max}]$
$(C; \mathbf{w}^2; q^2)$	$\lambda^-(x)$	$\lambda^-(x)$	$\sum_{U \in U_C} e\check{r}r_U^{(i)}$	$]0, x_{max}]$
$(C; \mathbf{w}^3; q^3)$	$\lambda^-(x)$	$\lambda^+(x)$	$\sum_{U \in U_C} c_U^{(i)}$	$]0, x_{max}]$

Table 4.3: Settings of the weighted voting games $(C; \mathbf{w}^j; q^j)$.

it then holds that $\Delta_k(S) = 1$ if a_k is a swing player for S, and $\Delta_k(S) = 0$ otherwise. Thus, the Shapley value of an agent is the higher, the more often it is pivotal for a subset S to meet the defined quotas. As we guaranteed for each weighted voting game $(C; \mathbf{w}^j; q^j)$ that the whole coalition wins, it also holds for the 3-vector weighted voting game that $v_g(C) = 1$. Because of the efficiency axiom, the resulting Shapley values $\phi_k^{(i)}(C, v_g)$ thus add up to one and can be interpreted as fair percentages in terms of Equation 4.45. Solving the latter finally yields the share x_k of each agent a_k and thus the desired distribution vector $\mathbf{x}_C$ specifying the payoff distribution of coalition C.

5 Evaluation

Having described DYCE in the previous chapter, in the following we provide a comprehensive evaluation of the proposed approach based on deterministic computer simulations. To this end, we define research questions *RQ* and conduct corresponding experiments *E-RQ* which allow us to analyze its specific characteristics and thus to support or reject the hypotheses defined at the beginning of this thesis in Section 1.3. Table 5.1 gives an overview of the whole evaluation including the approaches applied for experimentation. While selected research questions are examined using the OFAT approach because of a restricted number of relevant input parameters, the main part of the evaluation is based on DOE exploiting its powerful techniques as discussed in Section 2.3.
We start our investigations in Section 5.2 by analyzing the local performance of DYCE in terms of its ability to maximize the expected utility of an agent. To this end, we examine at which optimum rate or_{OSS_U} a maximal utility can be identified by the COPE algorithm described in Section 4.1.2 (*E-RQ*1). Moreover, we investigate how the utility depends on the number of operation schedules which are drawn from the operation schedule space of a unit as reflected by mean utility rate $\overline{ur}$ (*E-RQ*2). In Section 5.3, we then study the global performance and efficiency of DYCE using techniques from DOE. More precisely, we first use a full factorial design in order to determine the mean coalition values $\overline{v(C)}$ which are achieved in the course of coalition formation (*E-RQ*3) and examine the percentages of coalitions which are able to successfully fulfill their target product as quantified by fulfillment rate fr (*E-RQ*4). Afterwards, we use the same design in order to conduct a screening of the considered input parameters and investigate their effects on the global performance, computational cost, communication cost, and runtime (*E-RQ*5). These evaluation criteria are measured by means of the mean coalition value ($\overline{v(C)}$), the number of function calls of the CCOR algorithm (fc), the number of messages sent between the agents (msg), and the simulated realtime (sr). The determined effects allow us

section	experiment	approach	addressed hypothesis	measure
5.2	*E-RQ*1	OFAT	1.1 (local performance)	or_{OSS_U}
5.2	*E-RQ*2	OFAT	1.1 (local performance)	$\overline{ur}$
5.3	*E-RQ*3	DOE	1.2 (global performance)	$\overline{v(C)}$, fr
5.3	*E-RQ*4	DOE	1.2 (global performance)	$\overline{s_{C,n}}$
5.3	*E-RQ*5	DOE	1.2 (global performance)	$\overline{v(C)}$
			1.3 (efficiency)	fc, msg, sr
5.3	*E-RQ*6	DOE	1.2 (global performance)	$\widehat{\overline{v(C)}}$
			1.3 (efficiency)	$\widehat{fc}$, $\widehat{msg}$, $\widehat{sr}$
5.3	*E-RQ*7	OFAT	1.2 (global performance)	$\overline{v(C)}$
			1.3 (efficiency)	fc, msg, sr

Table 5.1: Overview of the evaluation.

to identify the most important factors and model the response surface of the considered criteria based on polynomial functions which provide corresponding predictions $\widehat{v(C)}$, $\widehat{fc}$, $\widehat{msg}$, and $\widehat{sr}$ (*E-RQ*6). We conclude the evaluation in Section 5.3.4 by investigating the applicability of DYCE to scenarios comprising large numbers of agents. We examine this aspect by again considering the four criteria global performance, computational cost, communication cost, and runtime. In order to draw conclusions from the investigations, Section 5.2.4 and 5.3.5 sum up experiments *E-RQ*1-*E-RQ*2 and *E-RQ*3-*E-RQ*7, respectively, and discuss the gained results in view of the initially stated hypotheses.
As can be seen from the above descriptions, we do not analyze economic aspects in terms of the final payoffs or long-run profits which agents gain from coalition formation. In order to draw reliable conclusions, this would require to model their trading strategies as well as the assumed market which is out of scope of this thesis.

5.1 Simulation System

In the following, we briefly describe the simulation system which is used for the upcoming investigations.
Generally, all conducted experiments are based on computer simulations. A single *simulation*, also referred to as *simulation run*, is given by an algorithm which is executed by the *simulation system*. The algorithm is configurable through a *parameterization* setting its *input parameters* (or *factors*) to specific values. The results of a simulation are measurable through defined *responses* which allow to quantify specific characteristics of interest. A simulation is executed on the basis of a *simulation scenario* which represents an examined real world setting being specified by means of concepts of DYCE-FM. An *experiment* consists of one or more simulations which are performed in order to answer an initially defined *research question*. The experimental results are evaluated through defined *measures* quantifying the quality of the outcomes based on the response values of the executed runs.
As general approach, we use discrete-event simulations for experimentation. I.e., a single simulation consists of a number of successive simulation steps each representing a simulated realtime interval of 100 ms. Within this time frame, all agents are able to perform their actions and send messages to their communication partners. A simulation ends if there are no more messages left to be transferred by the simulation system. Throughout the experiments, we generally use defined seeds for random number generation in order to guarantee a deterministic runtime behavior.

5.2 Local Performance

In the first part of the evaluation we examine the local performance of DYCE in terms of its ability to maximize the expected utility of an agent. As described in Section 4.1.2, the optimization problem is solved in the course of product portfolio management through application of the COPE algorithm which identifies a utility optimizing pair of operation schedule os_u and template portfolio TP forming the basis for the specification of a

product portfolio PP. In the upcoming sections, we thus conduct two experiments which allow us to answer the following research questions:

*RQ*1 Given an operation schedule os_U of a unit U and a template catalogue P_{tmp} with price predictions $C_{P_{tmp}}$, at which rate can an optimal utility $utility(TP, os_U)^*$ be identified by COPE?

*RQ*2 Given an operation schedule space OSS_U of a flexible unit U and a template catalogue P_{tmp} with price predictions $C_{P_{tmp}}$, how well can a maximum achievable utility $utility(TP, os_U)^+$ be approximated by COPE subject to the number of operation schedules os_U drawn from OSS_U?

The above research questions address two different properties of the COPE algorithm. Assuming a template catalogue P_{tmp} with price predictions $C_{P_{tmp}}$ for the comprised product templates p_{tmp} as well as an operation schedule os_U drawn from the schedule space OSS_U of a unit U, the first question asks for the rate at which COPE is able to identify an optimal utility $utility(TP, os_U)^*$. I.e., we are interested in the rate at which the algorithm is able to identify an optimal consistent template portfolio TP^*[1] given a single operation schedule os_U. We examine aspects of optimality using a single schedule because the schedule space of a flexible unit may be infinitely large in which case an optimal solution is not assessable. Contrary, in the context of research question *RQ*2 we explore how well a maximum achievable utility $utility(TP, os_U)^+$ can be approximated by COPE subject to the number of operation schedules os_U drawn from the schedule space OSS_U of a flexible unit U. Note that in this case the maximum achievable utility is not necessarily the optimal one but the best possible which can be obtained by applying COPE. I.e., we are here not interested in aspects of optimality (which are addressed by *RQ*1) but in the question of how the performance of COPE depends on the number of drawn schedules.
Before we describe experiment *E-RQ*1 and *E-RQ*2 which give answers to the above questions, we first specify the scenario which is used as basis for the executed simulations and detail the process which is applied in the context of *E-RQ*1 in order to generate optimal reference solutions.

5.2.1 Simulation Scenarios and Generation of Reference Solutions

In the following we describe the assumptions which are made with regard to the simulation scenarios being applied throughout the upcoming investigations. Moreover, we detail the approach which is used in order to generate optimal reference solutions in the context of experiment *E-RQ*1.
As general setting we suppose an agent which participates in the EPEX SPOT day-ahead auction and uses the COPE algorithm in order to identify a pair of template portfolio and operation schedule which maximizes its utility. Table 5.2 gives an overview of the applied simulation scenario by listing the relevant concepts of DYCE-FM along with the

[1] Recall from Definition 3.39 that a consistent template portfolio comprises product templates which product horizons are disjunct and cover the whole planning horizon.

concept	setting
T_{pl}	$t_{bu} = 1$ min, $t_u = 1$, $\Delta t = 15$, $t_{max} = 1440$, i.e. $T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq 95\}$
U	WEP ($RQ1$); CHP plant, lithium-ion battery ($RQ2$)
P_{tmp}	2013 catalogue of the EPEX SPOT day-ahead auction

Table 5.2: Scenario settings for research questions $RQ1$ and $RQ2$.

corresponding settings. The planning horizon T_{pl} covers the period of one day which is divided into 96 planning intervals of 15 minutes length. The unit U which is controlled by the agent differs according to the investigated problem. As in the context of experiment *E-RQ*1 we examine the algorithm's performance with regard to a single operation schedule only, U is given by an inflexible wind energy plant with a power of 2 MW which operation schedule space is thus inherently restricted. Contrary, the simulations of experiment *E-RQ*2 are based on two different units, i.e. a 4.7 kW CHP plant and a lithium-ion battery with a power of 5 kW for charging/discharging and a capacity of 10 kWh. Generally, the operation schedule spaces OSS_U of all units are independent and created on the basis of simulation models which implement their operational characteristics in a realistic way. More precisely, the schedule space of the WEP is created by running the simulation model for the considered time period of one day and including the output schedule os_U. Contrary, the schedule spaces of the flexible units are created by running the models n times using a random schedule as input and including those schedules which represent their actual operational behavior (as the units may deviate from the random schedules due to their operational constraints). As schedule costs we generally use the marginal costs of the units. More precisely, as the wind energy plant relies on renewable energy, we assume schedule cost of 0 €/kWh, i.e. $\forall t_{pl}^{(i)} \in T_{pl} : c_U^{(i)} =$ 0 €/kWh. With regard to the CHP plant, we generally face the problem that in the experiment we use historic wholesale prices of the EPEX SPOT in order to determine the price predictions of the product templates $p_{tmp} \in P_{tmp}$ which are used by the COPE algorithm as basis for building a template catalog TP. These prices are typically lower than the current marginal cost of a CHP plant which is not yet competitive with the cost of a large-scale power plant. Using the incompatible values in the simulations would thus lead to biased results as they are used by COPE in order to assess the utility of a template portfolio TP (cf. Equation 4.2). In the context of experiment *E-RQ*2, we therefore assume that the CHP plant is part of a coalition which is able to purchase fuel at wholesale prices and allows for a competitive marginal cost of 0.06 €/kWh. Contrary, for the battery we determine the cost based on the prices which are paid for the electricity required to charge the unit. Thus, we assume for each operation schedule that the unit's initial charging level is 0 kWh and it first consumes electrical energy. This way, the marginal cost can be calculated dynamically in the course of a simulation when determining the utility of a template portfolio according to Equation 4.2.

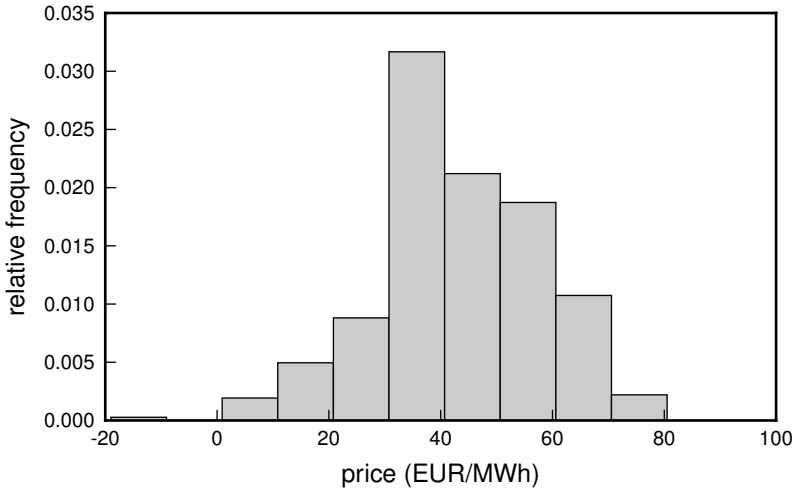


Figure 5.1: Price distribution of the 'Peakload' product.

Finally, with regard to the template catalogue P_{tmp} which is used as basis for building the template portfolio, we use the catalogue of the EPEX SPOT day-ahead auction of the year 2013. As already discussed in Section 4.1.1, this comprises 41 product templates which results in a search space of $2^{41} = 2\,199\,023\,255\,552$ template combinations. Because in this case it is impossible to find an optimal combination through exhaustive search, the catalogue provides an appropriate basis for our investigations. The base catalog $P_{tmp,base}$ which is required as further input by the COPE algorithm is thus given by the 24 single hour templates shown in the upper part of Figure 4.2. With regard to the price predictions $C_{P_{tmp}}$ for the product templates $p_{tmp} \in P_{tmp}$, it is generally necessary to use realistic values as these have consequences on the quality of the results. Thus, in the course of the following investigations, we determine all predictions $c^{(i)}_{p_{tmp}} \in C_{P_{tmp}}$ randomly according to product-related frequency distributions extracted from the EPEX SPOT day-ahead auction prices of the year 2013. As an example, Figure 5.1 depicts the frequency distribution of the block product 'Peakload', i.e. the relative frequency of the market prices at which the product was traded throughout the year. As shown, the most frequent prices lay between 30,76 € and 40,70 €, whereas the least frequent ones lay between -18.99 € and $-9,04$ €.

In order to examine aspects of optimality in experiment *E-RQ*1, it is necessary to generate optimal solutions which can be used as references for the actual experimental results. As described above, in the context of research question *RQ*1 we generally consider a single operation schedule os_U and search for a consistent template portfolio TP such that $utility(TP, os_U)$ is maximal. In order to create an optimal portfolio TP^* as reference solution, we first require that all electricity amounts of schedule os_U are positive, i.e. $\forall t^{(i)}_{pl} \in T_{pl} : e^{(i)}_U > 0$. An example for a valid schedule is shown in Figure 5.2. As

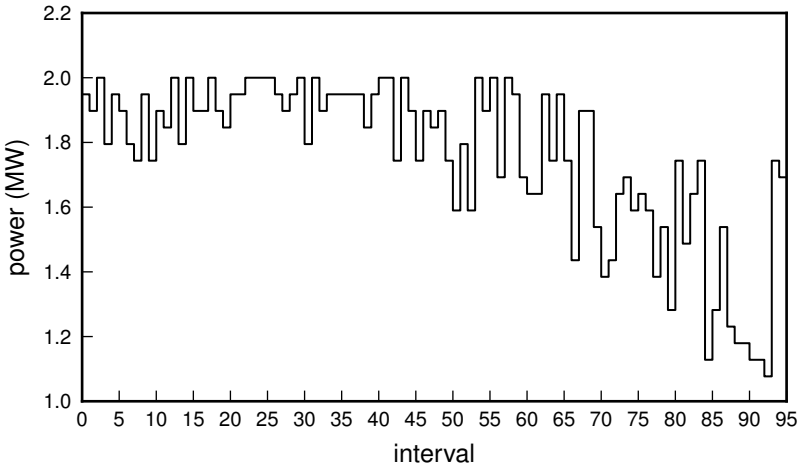


Figure 5.2: Exemplary operation schedule of a wind energy plant.

COPE optimizes utility, this requirement guarantees that the algorithm prefers product templates with higher price predictions to those with lower ones. We are then able to generate an optimal solution by creating a consistent template portfolio TP^* based on template catalog P_{tmp} which provides a unique maximal utility $utility(TP, os_U)^*$. In other words, we are able to create the one of the 2^{41} portfolios for which we know that it provides maximum utility with regard to a given operation schedule os_U. This is achieved by first assigning a random price prediction $c^{(i)}_{p_{tmp}}$ to each product template $p_{tmp} \in P_{tmp}$ according to the above described frequency distributions. Assuming that $c^{(i)}_{p_{tmp},max}$ is the maximum of all distributed values, we then create TP^* by randomly determining a consistent template portfolio and assigning the value of $c^{(i)}_{p_{tmp},max} + \varepsilon$ as price prediction to all comprised templates. Because all electricity values $e^{(i)}_U$ of os_U are positive, adding ε to $c^{(i)}_{p_{tmp},max}$ guarantees that the templates are the most beneficial to choose and thus form an optimal portfolio. To give a simple example, suppose we distribute random price predictions to all templates $p_{tmp} \in P_{tmp}$, determine the 'Baseload' template as consistent portfolio TP (as it covers all planning intervals $t^{(i)}_{pl} \in T_{pl}$), and finally assign maximum price prediction $c^{(i)}_{p_{tmp},max} + \epsilon$ to it. If COPE identifies this unique portfolio in the course of a simulation, it has found the optimal solution from the search space of all possible solutions.

5.2.2 *E-RQ*1: Optimum Rate

In order to examine the first of the initially defined research questions, in the following we conduct an experiment which allows us to assess the rate at which an agent is able to maximize its expected utility based on the COPE algorithm. A single simula-

ID	factor	description	min	max
X_1	OSS_U	operation schedule space	-	-
X_2	$C_{P_{tmp}}$	price predictions	-18.99	80.5
X_3	$drawnOS$	number of drawn operation schedules	1	1

Table 5.3: Parameterization of simulations of experiment $E\text{-}RQ1$.

tion run in the experiment is thus given by a single execution of COPE. In addition to the scenario settings specified in Table 5.2, Table 5.3 lists the input parameters which are varied throughout the experiment within the shown ranges. The operation schedule space OSS_U and price predictions $c^{(i)}_{p_{tmp}}$ for the product templates $p_{tmp} \in P_{tmp}$ are generated as described in the previous section. In accordance with our experimental goal, the operation schedule space of the assumed wind energy plant inherently restricts the number of drawn schedules $drawnOS$ to a single schedule in each simulation run. Based on the shown parameterization, the experiment is executed as follows:

$E\text{-}RQ1$ For each of 60 different operation schedule spaces OSS_U of an inflexible producer, a series of 1000 simulations is executed, where in each run the price predictions $c^{(i)}_{p_{tmp}}$ for the product templates $p_{tmp} \in P_{tmp}$ are randomly determined according to specific distributions and the resulting template portfolio TP is captured as response.

I.e., for a single operation schedule space (and thus operation schedule), we execute the COPE algorithm 1000 times with varying price predictions as input and capture the identified template portfolios TP as output which we then compare to the optimal reference solutions TP^*. In order to increase the reliability of the results, we conduct this series for 60 different schedule spaces which we obtain by simulating the operational behavior of the WEP over a year and extracting those schedules which fulfill the previously discussed requirement of strict positivity.
Given this experimental setup, we are finally able to assess the performance of COPE by defining a schedule space-related optimum rate or_{OSS_U} which is the quotient of the number of optimal template portfolios $count(TP^*)$ and the number of simulation runs $count(r)$ which were executed with schedule space OSS_U as input, i.e.

$$or_{OSS_U} = \frac{count(TP^*)}{count(r)}. \tag{5.1}$$

From the above equation it follows that $or \in [0, 1]$, where a rate of 0 means that COPE identified none optimal portfolio at all and a rate of 1 means that it found all optimal portfolios possible. As specified above, the number of simulations $count(r)$ performed for each schedule space OSS_U is set to 1000 runs.
The results of experiment $E\text{-}RQ1$ are shown in Figure 5.3 in the form of two boxplots depicting the 60 obtained optimum rates on two different scales. Covering the whole interval $[0, 1]$, the plot on the left reveals that we generally obtain very good results, with a minimum optimum rate of 0.981 and a maximum rate of 0.996. The plot on

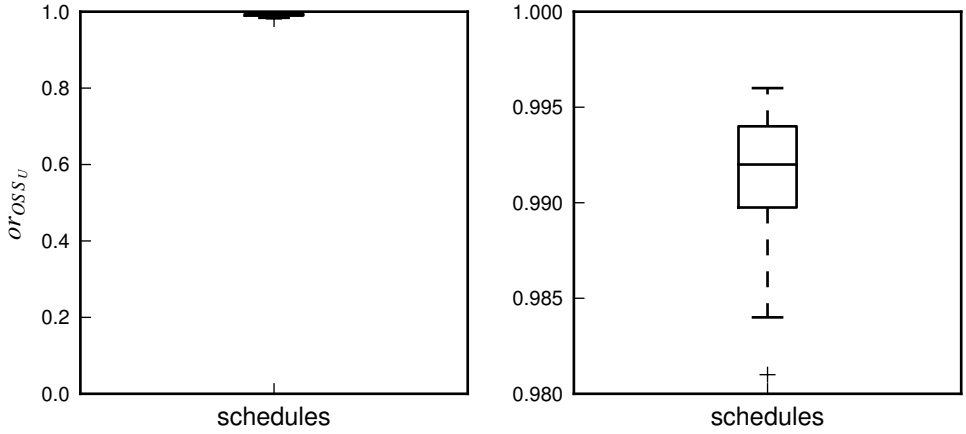


Figure 5.3: Optimum rates or_{OSS_U} for 60 operation schedule spaces.

the right provides a more detailed view of the outcomes by depicting the quartiles of the 60 rates via the box and the range of the data through the upper and lower whisker, where an outlier is given by the point which extends the latter. From a lower quartile of 0.99 it follows that three-quarter of the results show an optimum rate of 0.99 or higher, i.e. for each operation schedule of the corresponding schedule spaces COPE identified an optimal template portfolio in at least 99 % of the 1000 executed runs. Moreover, the narrow range of the data suggests a high reliability of the results even when taking outliers into account.

5.2.3 *E-RQ2*: Mean Utility Rate

In order to answer research question *RQ2*, in the following experiment we examine how well a maximum achievable utility $utility(TP, os_U)^+$ can be approximated by the COPE algorithm subject to the number of operation schedules drawn from the schedule space of a flexible unit. Thus, a single simulation in the experiment is again given by a single execution of COPE. However, in contrast to our previous investigations, we are now not interested in aspects of optimality but in the algorithmic performance with regard to a best achievable solution which can be obtained through the application of COPE. More specifically, we want to analyze how the quality of the results depends on the number of operation schedules drawn from the schedule space of a unit. As the assumed CHP plant and lithium-ion battery differ in their operational capabilities and thus in their degree of flexibility, they allow for a sound analysis of the considered problem. In this regard, Figure 5.4 shows a comparison of the units' operation schedule spaces OSS_U which are used in the following simulations, where $|OSS_U| = 100\,000$ and each included operation schedule os_U covers 96 planning intervals as determined by planning

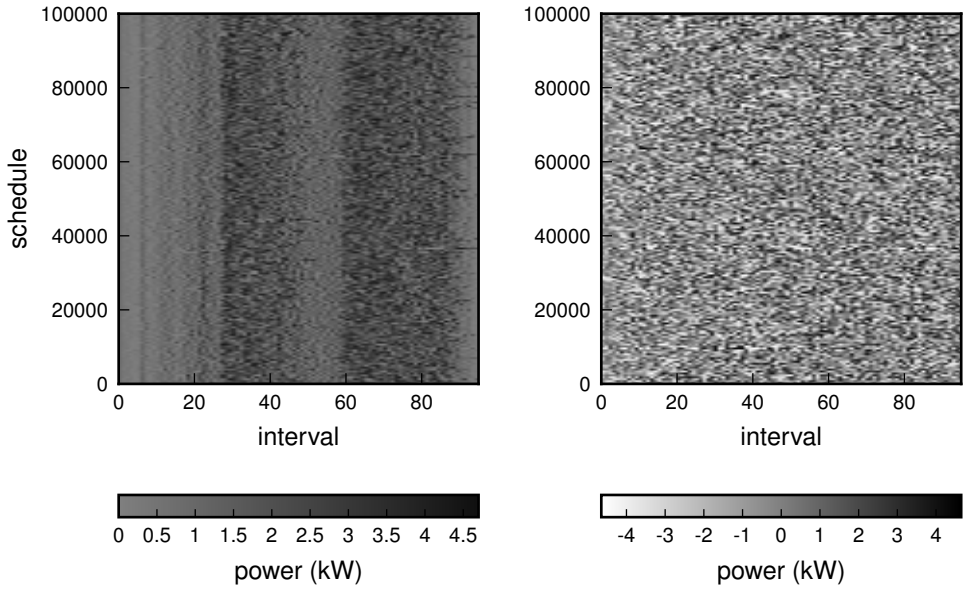


Figure 5.4: Schedule space of the applied CHP plant (left) and battery (right).

horizon T_{pl}. As can be seen from the different depths of the colors, the schedules of the battery provide a higher degree of operational flexibility by covering a wider scope of power values. With regard to our considered problem, we are particularly interested in the question if this fact has an impact on the number of schedules which has to be drawn from the schedule space because a more flexible schedule space generally contains more schedules associated with a lower utility.
The parameterization of the executed simulations is shown in Table 5.4. Besides the operation schedule space OSS_U and the price predictions $c^{(i)}_{p_{tmp}}$ for the product templates $p_{tmp} \in P_{tmp}$, we now also vary the number of operation schedules which are randomly drawn from OSS_U within the specified ranges. The schedule space and the price predictions are again generated according to the approaches described in Section 5.2.1. Based on the shown parameterization, the experiment is conducted as follows:

*E-RQ*2 For the operation schedule space OSS_U of a flexible producer and a storage, with $|OSS_U| = 100\,000$, ten simulations are executed throughout which the number of drawn operation schedules *drawnOS* is successively increased in steps of 10 000 and the final utilities $utility(TP, os_U)$ are captured as responses. This series is repeated 1000 times with varying price predictions $c^{(i)}_{p_{tmp}}$ for the product templates $p_{tmp} \in P_{tmp}$ which are randomly determined according to specific distributions.

More specifically, for a given operation schedule space we execute a series of ten simulations in which we increase the number of drawn schedules in steps of 10 000, starting

ID	factor	description	min	max
X_1	OSS_U	operation schedule space	-	-
X_2	$C_{P_{tmp}}$	price predictions	−18.99	80.5
X_3	*drawnOS*	number of drawn operation schedules	10 000	100 000

Table 5.4: Parameterization of simulations of experiment *E-RQ*2.

from 10 000 up to the limit of 100 000 schedules. As response of each run, we capture the utility of the identified template portfolio and operation schedule which was achieved drawing *drawnOS* schedules. In order to increase the reliability of the results, we repeat this series 1000 times with varying price predictions.
Based on this experimental setup, the quality of a single simulation is assessed by means of a utility rate *ur* which is defined as the quotient of the achieved utility $utility(TP, os_U)$ and the maximum achievable utility $utility(TP, os_U)^+$, i.e.

$$ur = \frac{utility(TP, os_U)}{utility(TP, os_U)^+}. \tag{5.2}$$

The maximum achievable utility $utility(PP, os_U)^+$ which is obtainable through application of the COPE algorithm can be assessed by exploring the whole schedule space OSS_U (i.e. by setting *drawnOS* to 100 000). Given *ur*, we are finally able to assesses the quality of a series of n simulations by calculating the mean of the corresponding utility rates, i.e.

$$\overline{ur} = \frac{\sum_{i=1}^{n} ur_i}{n}. \tag{5.3}$$

In the following, we use the above measure in order to determine the average rate of the $n = 1000$ simulations which are executed with the number of drawn schedules being equal. Generally, it holds that both $ur \in [0, 1]$ and $\overline{ur} \in [0, 1]$.
The results of experiment *E-RQ*2 are shown in Figure 5.5, where the upper plots pertain to the CHP plant and the lower ones to the battery. Each plot describes the mean utility rates $\overline{ur}$ for the different numbers of drawn operation schedules *drawnOS* which were calculated over the 1000 repetitions, where the bar charts cover the whole interval $[0, 1]$ and the line charts provide a more detailed view of the outcomes by restricting the scale to interval $[0.9, 1]$. As shown, we obtain very good results for both the CHP plant and the battery, with minimum mean utility rates of 0.99628 and 0.95072 and maximum rates of 0.99994 and 0.99774, respectively[2]. Generally, $\overline{ur}$ increases with an increasing number of drawn schedules which is reasonable as we successively explore higher fractions of the search space and thus raise the probability to find more beneficial schedules. Moreover, the results show that even for the small numbers of drawn schedules COPE yields high utility rates for both units, where the values of the CHP plant are close to

[2] Because the rates for 100 000 drawn schedules are necessarily 1, we leave them out of discussion.

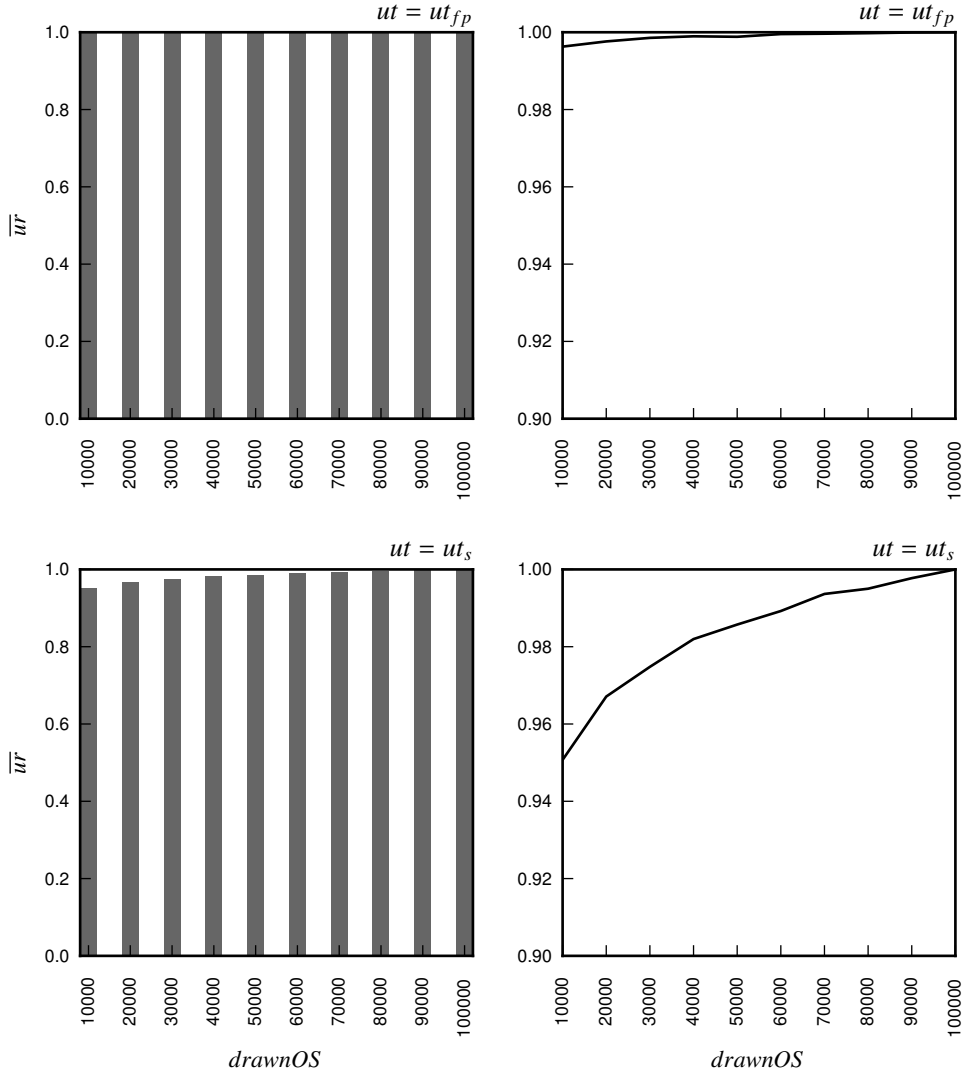


Figure 5.5: Mean utility rates $\overline{ur}$ for a CHP plant (top) and a battery (bottom).

the maximum achievable solution. With regard to the different operational capabilities of the units, the results finally indicate that for a more flexible unit a higher number of schedules has to be drawn from the schedule space in order to obtain results comparable to that of a less flexible unit. The battery yields lower rates for all numbers of drawn schedules, where the difference is the greater, the less schedules are drawn. However, taking into account the great differences in the operational flexibilities, the differences in the utility rates can still be considered very small. Finally, it should be noted that the

algorithm's worst case complexity of $O(m \cdot \log m + n \cdot m)$ allows to evaluate very large numbers of drawn schedules which may easily exceed the amounts investigated in the experiment.

5.2.4 Summary and Discussion

Before evaluating the global performance and efficiency of DYCE, in the following we first summarize the previous experiments and draw conclusions with regard to the local performance of the approach based on the obtained results. To this end, we reconsider Hypotheses 1.1 which we specified at the beginning of the thesis in Section 1.3 in order to capture our related expectations. This was formulated as follows:

> Through integration of an appropriate optimization algorithm, a distributed aggregation method allows actors to successfully optimize their expected utility and approximate the optimal solution of the corresponding local optimization problem to a high degree.

In order to examine the hypothesis, we conducted experiment *E-RQ*1 and *E-RQ*2 which evaluated the performance of the COPE algorithm being applied by agents in the course of product portfolio management in order to optimize their local utility. As there was only a small number of factors to be handled, we chose the OFAT approach for experimentation. Goal of experiment *E-RQ*1 was the assessment of optimum rate or_{OSS_U} reflecting the frequency at which COPE is able to identify an optimal template portfolio TP^* providing an optimal utility $utility(TP, os_U)^*$ given a considered operation schedule os_U. The simulations revealed very good results showing that in case of three quarter of the 60 examined schedules an optimal portfolio could be identified in at least 99 % of the 1000 runs which were executed for each schedule. Experiment *E-RQ*2 then examined the extent to which the algorithm's performance depends on the number of operation schedules drawn from the operation schedule space of a unit. As in practice the degree of flexibility typically varies among units of different type, we compared the schedule space of a CHP plant and a lithium-ion battery in the course of the investigations. As quality measure, we this time considered the mean utility rate $\overline{ur}$ which is defined as the average ratio between the achieved and the maximum achievable utility calculated over a specific number of simulation runs, with 1 being the maximum achievable rate. Generally, the obtained results revealed a very good performance with regard to both the CHP plant and the battery, with lowest mean utility rates of 0.99628 and 0.95072 in case of 10 000 and highest rates of 0.99994 and 0.99774 in case of 90 000 drawn schedules. The outcomes also showed that a higher degree of flexibility requires a higher number of schedules to be drawn from the schedule space of a unit in order to achieve results comparable to those of a less flexible unit. However, the differences in the obtained rates were still surprisingly small compared to the relatively high difference in the units' degree of flexibility.
All in all, we can conclude that the outcomes of experiment *E-RQ*1 and *E-RQ*2 support our initially stated hypothesis by showing that the COPE algorithm is suitable for

optimizing the expected utility of an agent and approximating the maximum of the corresponding optimization problem to a high degree. While the received results indicated that high utilities can already be achieved with small numbers of drawn operation schedules, the algorithm's worst case complexity of $O(m \cdot \log m + n \cdot m)$ generally allows to process large amounts of samples making the heuristic both an effective and efficient approach for local optimization.

5.3 Global Performance and Efficiency

Having analyzed the local performance of DYCE, in the following we investigate its global performance and efficiency by means of a comprehensive study based on techniques from the domain of Design of Experiments. Moreover, we examine its applicability to scenarios comprising large numbers of agents. The corresponding research questions are defined as follows:

*RQ*3 Given a power grid G with units $\widetilde{U}$ and a target product p_{tar} for which coalitions to form, how well can an optimal mean coalition value $\overline{v(C)} = 1$ be approximated by DYCE and at which rate are coalitions able to fulfill p_{tar}?

*RQ*4 Given a power grid G with units $\widetilde{U}$ and a target product p_{tar} for which coalitions to form, which distances cover the scopes s_C of the resulting coalitions C?

*RQ*5 With regard to the criteria global performance, computational cost, communication cost, and runtime, what is the impact of the different factors and factor interactions on the respective responses of DYCE?

*RQ*6 With regard to the criteria global performance, computational cost, communication cost, and runtime, what is the algorithmic behavior of DYCE subject to the most influential factors, i.e. what do the respective response surfaces look like?

*RQ*7 Given a power grid G with units $\widetilde{U}$ and a target product p_{tar} for which coalitions to form, how does DYCE scale with an increasing number of agents A?

While we examine research questions *RQ*3-*RQ*6 based on DOE, *RQ*7 is answered using the OFAT approach because of a limited number of relevant factors. In the context of *RQ*3, we first investigate one of the most crucial characteristics of DYCE by analyzing its ability to approximate an optimal global solution. The latter is given by a coalition structure CS^* for which the mean coalition value is maximal. We thus examine how well the formed coalitions are able to approximate the values of their pursued target product on average. As further indicator for the global performance, we determine the percentage of coalitions which is finally able to fulfill the target product. In the context of research question *RQ*4, we then examine the effectiveness of neighborhood formation by assessing the scopes of the coalitions $C \in CS$ in terms of the maximum distances which exist between their units. As in the simulations we generally allow coalitions to span the whole grid, we are interested in the question of how the distance-based neighborhood

formation influences the scope if no upper limit is provided. In the context of $RQ5$ and $RQ6$, we then take a closer look at the algorithmic behavior of DYCE with regard to the evaluation criteria global performance, computational cost, communication cost, and runtime. In terms of DOE, these can be considered as responses of the algorithm and thus examined by means of experimental designs. In the context of $RQ5$, we first conduct a factor screening by initially assessing the effects of the relevant factors and their two-way interactions on the four criteria. We restrict the discussion to main effects and two-way interactions as there are $2^{14} - 1 = 16\,383$ effects assessable for each criterion given the 14 considered factors of DYCE which are analyzed on the basis of a full factorial design. Thus, we refer to the sparsity of effects principle which states that in most systems only 20 % of the main and two-way interaction effects are of significance while the other two-way and higher-order interaction effects vary to the extent of the normal error [AW07]. Assessing the effects allows us to screen out the most important factors and investigate their influence on the considered responses in more detail. In $RQ6$, we thus examine the dependency between the most pivotal factors and the four evaluation criteria via response surface modeling. This allows us to provide a detailed description of the algorithmic characteristics of DYCE in the form of polynomial models which are created for all four responses. In the context of $RQ7$, we finally examine the scalability of the approach with regard to scenarios comprising large numbers of participating agents. To this end, we again use the above criteria for evaluation and investigate how these behave if neighborhoods can be extended either to the whole or only a limited part of the grid.

As in case of research question $RQ1$, studying aspects of optimality requires the generation of reference solutions which can be used in order to assess the output of the simulations. Before detailing the experiments and their results in Section 5.3.2-5.3.4, we thus start again by describing the used simulation scenarios as well as the approach applied for creating optimal coalition structures.

5.3.1 Simulation Scenarios and Generation of Reference Solutions

In the following we describe the simulation scenarios which are used as basis for the performed simulations as well as the procedure which is applied for the creation of optimal coalition structures CS^*. Because we intend to study scenarios with the number of agents being in the order of thousands, it is generally impossible to determine the latter through calculation of all possible solutions. Thus, we synthetically create the operation schedules of the units in the considered power grids in such a way that they allow for the formation of an optimal coalition structure which comprised coalitions C^* fulfill their targets product exactly, i.e. $\forall C^* \in CS^* : v(C^*) = 1$. As described in more detail below, this is achieved by assigning random numbers to the different schedule-related attributes in a systematic fashion.

The scenario settings which are used throughout the upcoming examinations are shown in Table 5.5. We generally assume that agents form coalitions day-ahead and apply a minute-based planning horizon T_{pl} with a planning interval length of 60 min and a

concept	setting
T_{pl}	$t_{bu} = 1$ min, $t_u = 1$, $\Delta t = 60$, $t_{max} = 1440$, i.e. $T_{pl} = \{t_{pl}^{(i)} \mid 0 \leq i \leq 23\}$
G	power grid with synthetic topology and line impedances
$\widetilde{U}$	producers (ut_p) with synthetic operation schedule spaces
P_{tmp}	singleton template catalogue $\{p_{H12}\}$

Table 5.5: Scenario settings for research questions $RQ3$ to $RQ7$.

horizon length of 1 day (i.e. $t_{max} = 1440$). The power grid G is generated synthetically according to the parameterization of the given simulation as described in more detail below. The same holds for the set of connected units $\widetilde{U}$ which are assumed to be of type producer. As we already examined the local performance of DYCE in Section 5.2, the applied template catalogue P_{tmp} is restricted to a single product template p_{H12} which specifies the constraints for the definition of a single hour product to be fulfilled at 12:00 pm on the considered day. Before we describe the method for the creation of G and $\widetilde{U}$ in more detail, we first justify our assumptions and intention to configure the operation schedules of the units through random numbers by arguing that the made choices are reasonable and have no impact on the quality of the experimental results.
With regard to the restricted template catalogue P_{tmp}, first recall from Section 4.1.2 that coalition formation for different products is conducted sequentially in disjunct time intervals. Thus, the size of the catalogue has no impact on the quality of the results. This also holds for the time of fulfillment which is prescribed by the included product template p_{H12} because we create the operation schedules of the units synthetically and independent from any temporal aspects which makes the results independent from temporal aspects as well. With regard to power grid G, recall from Definition 3.10 that agents form neighborhoods based on grid-related data but the formation process is still independent from technical aspects as neighborhoods are expanded by a specific number of agents and not a specific distance. I.e., after having converted the grid data into an ordered list of neighbors which is sorted by their distance, an agent conducts the activity of neighborhood formation and coalition formation without using the information about the grid again. Thus, assuming that all agents have access to the same data, the power grid G can not be expected to have an impact on the quality of the results. Finally, with regard to the connected units $\widetilde{U}$ and the chosen unit type ut_p, recall from Definition 3.25 that in the context of DYCE units are considered as abstract types, i.e. flexible/inflexible producers, flexible/inflexible consumers, and storage. These are independent from specific technological aspects and only differ with regard to their production and consumption behavior as well as their operational flexibility. As the operational flexibility is only exploited in the course of product portfolio management for local utility optimization and the production/consumption behavior only has an impact on the sign of the cumulative electricity amounts of the formed coalitions, the unit type also has no impact on the quality of the final results.

With regard to our intention to configure the operation schedules of the units through random numbers, we argue that this is a reasonable approach because in real settings the electricity amount $e_U^{(i)}$, error $err_U^{(i)}$, and cost $c_U^{(i)}$ of a unit in a planning interval $t_{pl}^{(i)}$ are determined by the unit's technology and possibly by further external conditions like varying weather forecasts or the local preferences of an agent. I.e., across the whole power grid these values typically vary between units because of different factors which we model by assigning random numbers to the different attributes of the schedules according to the approach described below.

Although the above arguments justify a synthetic generation of power grid G, we still take account of technical characteristics of real networks when creating the scenarios for the upcoming simulations. More precisely, for a single run we generate $G = (V_G, E_G)$ by creating a random radial network with a maximum branching factor of 3 and assigning random complex numbers as apparent impedances to the lines $\{v_{G,k}, v_{G,l}\} \in E_G$. The size of the grid is determined by the given number of units which in turn is determined by the number agents as specified by the parameterization of the simulation, where all units are assigned to the grid nodes $v_{G,i} \in V_G$ in a one-to-one relationship. The operation schedules os_U of all units U are then configured in such a way that they allow for an optimal solution, i.e. an optimal coalition structure CS^* with mean coalition value $\overline{v(C)} = 1$. To this end, we first define a target product p_{tar} based on product template $p_{H12} \in P_{tmp}$ which has to be eventually fulfilled by each coalition $C^* \in CS^*$ and which product horizon $T_{pr}^{(p_{tar})} = \{t_{pr}^{(12)}\}$ complies with p_{H12} and planning horizon T_{pl}. The target electricity amount $e_{p_{tar}}^{(i)}$, target error $err_{p_{tar}}^{(i)}$, and target cost $c_{p_{tar}}^{(i)}$ are generally determined by the parameterization of the given simulation. The target electricity amount $e_{p_{tar}}^{(i)}$ is used in order to assess the number of agents being included in an optimal coalition $C^* \in CS^*$ which is calculated as

$$|C^*| = \left\lfloor 0.1 \cdot e_{p_{tar}}^{(i)} \right\rfloor. \tag{5.4}$$

For example, a target electricity amount of 100 kWh results in a coalition size of 10 members. This approach ensures that agents always contribute similar quantities to a coalition as the target amount is distributed to a member count which is determined by the amount itself. If the total number of agents $|A|$ in a simulation is no multiple of the optimal coalition size $|C^*|$, we create n coalitions C^* and one coalition C_r^* such that $|A| = n \cdot |C^*| + |C_r^*|$. As the electricity amount determines the optimal coalition size, it also has an impact on the number of optimal coalitions being part of an optimal coalition structure CS^* which is calculated as

$$|CS^*| = \left\lfloor \frac{|A|}{|C^*|} \right\rfloor. \tag{5.5}$$

In order to achieve that all optimal coalitions exactly fulfill p_{tar}, we next create a cumulative contribution $con_{C^*,p_{tar}}$ for each $C^* \in CS^*$ which cumulative values exactly match the specified target values. Thus, for the given product interval $t_{pr}^{(i)} \in T_{pr}^{(p_{tar})}$, with $i = 12$, it holds that

$$con_{C^*,p_{tar}}(t_{pr}^{(i)}) = (e_{C^*}^{(i)}, err_{C^*}^{(i)}, c_{C^*}^{(i)}) = (e_{p_{tar}}^{(i)}, err_{p_{tar}}^{(i)}, c_{p_{tar}}^{(i)}). \tag{5.6}$$

In order to make the aggregation of the agents' individual contributions $con_{p_{tar}}(t_{pr}^{(i)})$ equal cumulative contribution $con_{C^*,p_{tar}}(t_{pr}^{(i)})$, we assign appropriately calculated random numbers to the operation schedules os_U of the controlled units U in order to specify the electricity amounts $e_U^{(i)}$, errors $err_U^{(i)}$, and costs $c_U^{(i)}$, taking into account Equation 4.27-4.29 for the calculation of the cumulative values $e_C^{(i)}$, $err_C^{(i)}$, and $c_C^{(i)}$. To recapitulate, for a product interval $t_{pr}^{(i)}$, the latter are calculated as

$$e_C^{(i)} = \sum_{U \in U_C} e_U^{(i)}, \tag{5.7}$$

$$err_C^{(i)} = \frac{\sqrt{\sum_{U \in U_C} MSE_U^{(i)}}}{|e_C^{(i)}|}, \tag{5.8}$$

$$c_C^{(i)} = \frac{\sum_{U \in U_C} c_U^{(i)} \cdot e_U^{(i)}}{|e_C^{(i)}|}. \tag{5.9}$$

Taking into account Equation 5.6, it follows from Equation 5.7 that we can distribute the target electricity amount $e_{p_{tar}}^{(i)}$ to all units $U \in U_{C^*}$ of an optimal coalition C^* provided that $\sum_{U \in U_{C^*}} e_U^{(i)} = e_{p_{tar}}^{(i)}$. To this end, we assign random numbers $e_U^{(i)}$ to the corresponding schedules os_U such that the equation is met.
With regard to the error, we can rewrite Equation 5.8 given Equation 5.6 as

$$\begin{aligned} err_C^{(i)} &= \frac{\sqrt{\sum_{U \in U_C} MSE_U^{(i)}}}{|e_C^{(i)}|} \\ \Leftrightarrow err_{p_{tar}}^{(i)} &= \frac{\sqrt{\sum_{U \in U_C} MSE_U^{(i)}}}{|e_{p_{tar}}^{(i)}|} \\ \Leftrightarrow (err_{p_{tar}}^{(i)} * |e_{p_{tar}}^{(i)}|)^2 &= \sum_{U \in U_C} MSE_U^{(i)} \end{aligned} \tag{5.10}$$

According to the above equation, the individual errors of the units can be determined by first dividing the term $(err_{p_{tar}}^{(i)} * |e_{p_{tar}}^{(i)}|)^2$ randomly into $|C^*|$ shares in order to retrieve the mean squared errors $MSE_U^{(i)}$, where we require that these do not exceed the previously determined electricity amounts, i.e. $MSE_U^{(i)} < e_U^{(i)}$ for all $U \in U_{C^*}$. Dividing the roots of the mean squared errors by the amounts finally yields the desired errors $err_U^{(i)}$ (cf. Equation 4.26).

Finally, with regard to the cost, we can rewrite Equation 5.9 based on Equation 5.6 as follows:

$$\begin{aligned} c_C^{(i)} &= \frac{\sum_{U \in U_C} c_U^{(i)} \cdot e_U^{(i)}}{e_C^{(i)}} \\ \Leftrightarrow c_{p_{tar}}^{(i)} &= \frac{\sum_{U \in U_C} c_U^{(i)} \cdot e_U^{(i)}}{e_{p_{tar}}^{(i)}} \\ \Leftrightarrow c_{p_{tar}}^{(i)} \cdot e_{p_{tar}}^{(i)} &= \sum_{U \in U_C} c_U^{(i)} \cdot e_U^{(i)} \end{aligned} \tag{5.11}$$

This equation shows that the individual cost values $c_U^{(i)}$ cannot be determined without taking the electricity amounts $e_U^{(i)}$ into account. Assuming that $U_{C^*} = \{U_1, \ldots, U_n\}$ and all $e_U^{(i)} \geq 0$, $c_U^{(i)} \geq 0$ (because we consider units of type producer), we can first calculate the cost value of the nth unit U_n as follows:

$$\begin{aligned} c_{p_{tar}}^{(i)} \cdot e_{p_{tar}}^{(i)} &= \sum_{m=1}^{n} c_{U_m}^{(i)} \cdot e_{U_m}^{(i)} \\ \Leftrightarrow c_{p_{tar}}^{(i)} \cdot e_{p_{tar}}^{(i)} &= \sum_{m=1}^{n-1} c_{U_m}^{(i)} \cdot e_{U_m}^{(i)} + c_{U_n}^{(i)} \cdot e_{U_n}^{(i)} \\ \Leftrightarrow \frac{1}{e_{U_n}^{(i)}}((c_{p_{tar}}^{(i)} \cdot e_{p_{tar}}^{(i)}) - \sum_{m=1}^{n-1} c_{U_m}^{(i)} \cdot e_{U_m}^{(i)}) &= c_{U_n}^{(i)}. \end{aligned} \tag{5.12}$$

Given this cost value along with the above stated assumptions, we can finally choose the values $c_{U_1}^{(i)}$ to $c_{U_{n-1}}^{(i)}$ of the units U_1 to U_{n-1} randomly from the following intervals:

$$0 \leq c_{U_k}^{(i)} \leq \frac{1}{e_{U_n}^{(i)}}(c_{p_{tar}}^{(i)} * e_{p_{tar}}^{(i)}) \qquad \text{for } k = 1, \tag{5.13}$$

$$0 \leq c_{U_k}^{(i)} \leq \frac{1}{e_{U_n}^{(i)}}((c_{p_{tar}}^{(i)} * e_{p_{tar}}^{(i)}) - \sum_{m=1}^{k-1} c_{U_m}^{(i)} * e_{U_m}^{(i)}) \qquad \text{for } 1 < k \leq n-1. \tag{5.14}$$

Having determined the electricity amounts $e_U^{(i)}$, errors $err_U^{(i)}$, and costs $c_U^{(i)}$ for all units U, we are finally able to specify their operation schedules os_U by setting the entries $os_U(t_{pl}^{(i)})$ to the generated values which thus finally allow for the formation of an optimal coalition structure CS^*.

5.3.2 *E-RQ*3, *E-RQ*4: Global Optimum Approximation and Coalition Scopes

We next provide answers to research question *RQ*3 and *RQ*4 by examining the ability of DYCE to approximate an optimal mean coalition value $\overline{v(C)} = 1$ and determining the rate at which formed coalitions are able to fulfill their target products. Moreover, we

ID	factor	description	−1	1
X_1	$e^{(i)}_{p_{tar}}$	target electricity amount in kWh	100	200
X_2	$err^{(i)}_{p_{tar}}$	target error	0.01	0.03
X_3	$c^{(i)}_{p_{tar}}$	target cost in €/kWh	0.1	0.2
X_4	$e^{(i)}_{p_{tar},-/+}$	tolerance band for target electricity amount in kWh	0.8	8
X_5	$err^{(i)}_{p_{tar},-/+}$	tolerance band for target error	0.005	0.01
X_6	$c^{(i)}_{p_{tar},-/+}$	tolerance band for target cost in €/kWh	0.05	0.1
X_7	$\|A\|$	number of participating agents	1000	2000
X_8	r_{init}	rate of initiators	0.1	0.4
X_9	ext_N	extension count	10	40
X_{10}	$i_{u,max}$	number of consecutive unsuccessful initiations	2	8
X_{11}	$rCom$	max. number of proposals accepted at once	1	10
X_{12}	t_{bi}	time until responders try to become initiators in min	2	8
X_{13}	bi_{max}	max. number of role changes to become an initiator	2	8
X_{14}	oc_{max}	max. number of optimization cycles	1	40

Table 5.6: Parameterization of simulations of experiment *E-RQ*3, *E-RQ*4, and *E-RQ*5.

investigate the scopes of the coalitions in order to analyze the impacts of a distance-based formation of neighborhoods. For each research question, we conduct an experiment using a two-level full factorial design which we also apply for a factor screening in the context of *RQ*5 in Section 5.3.3. We choose this design type for the considered problems because it allows for a detailed and complete analysis by covering all possible factor level combinations (cf. Section 2.3.2.1). A single simulation in each experiment is given by a single execution of DYCE.

Table 5.6 gives an overview of the input parameters which are used in a simulation in order to configure different aspects of the activities neighborhood formation and coalition formation. The total number of 14 parameters results in a full factorial design comprising $2^{14} = 16\,384$ simulation runs in which each parameter is systematically varied between the listed low level (−1) and high level (1). Before describing the actual experiments and their results, we start by discussing the values which were chosen for the different levels. These were on the one hand determined by analyzing the considered problem of coalition formation in electricity markets and on the other hand by evaluating the results of test runs which were performed for the assessment of reasonable settings.

The first three factors of Table 5.6 represent the attributes of the target product p_{tar} which the agents are intended to fulfill in the course of coalition formation as described in the previous section. The tolerance bands of the attributes within which the target product is considered to be fulfilled are defined by the corresponding factors X_4-X_6. In order to limit the total number of parameters, all bands are specified symmetrically, i.e. the negative and positive tolerance values are equal and handled as a single factor. With regard to the first parameter, target electricity amount $e^{(i)}_{p_{tar}}$, the low level is set to

100 kWh which is the minimum volume currently tradable at the EPEX SPOT. Contrary, the high level is set to the double amount of 200 kWh. As the amount is used in order to determine the size of the optimal coalitions in a generated scenario (cf. Equation 5.4), the chosen values result in reasonable member counts of 10 and 20 agents as well as sensible numbers of coalitions in the optimal coalition structures as discussed in more detail below. Specifying considerably higher levels would yield numbers of coalitions being too low for a meaningful analysis. As defined by factor X_4, the tolerance band for the target electricity amount allows coalitions to deviate from the chosen values by ±0.8 kWh in case of the low and ±8 kWh in case of the high level.
As second attribute of target product p_{tar}, the target error $err^{(i)}_{p_{tar}}$ is varied between a restrictive low level of 0.01 and a more tolerant high level of 0.03. According to Equation 4.26, these settings correspond to a $RMSE$ of 1/3 kWh (low/high) for the target amount of 100 kWh and 2/6 kWh for the amount of 200 kWh. The related tolerance bands $(0.005, 0.005)$ and $(0.01, 0.01)$ allow for an absolute deviation of ± 0.5/1 kWh and ± 1/2 kWh.
As last attribute of the target product, target cost $c^{(i)}_{p_{tar}}$ is finally set to a low level of 0.1 €/kWh and a high level of 0.2 €/kWh. These values roughly reflect the marginal costs of small-scale units like CHP plants or batteries which are particularly expected to form coalitions. Note that some types of distributed generators like solar power or wind energy plants have marginal costs of 0 €/kWh. However, because we randomly distribute the target cost to the members of an optimal coalition when generating a scenario, using this value as low level would result in schedule cost of 0 €/kWh in which case agents would meet the target value even before coalition formation has actually started (cf. Equation 4.29). As specified by factor X_6, the tolerance band for the cost allows coalitions to deviate from the specified settings by ± 0.05 €/kWh in case of the low and ± 0.1 €/kWh in case of the high level.
Besides the parameters related to target product p_{tar}, factor X_7 defines the number of agents which participates in the market and thus the number of units which is connected to the grid. The chosen levels of 1000 and 2000 producers reflect typical quantities which occur in today's mid voltage grids including the connected low voltage grids. Moreover, taking into account the above discussed optimal coalition sizes of 10 and 20 members, these settings result in a sufficient number of coalitions in a generated optimal coalition structure CS^*. As shown in Table 5.7, for a given simulation the latter can consist of 50, 100, or 200 coalitions depending on the total number of agents $|A|$ and the target electricity amount $e^{(i)}_{p_{tar}}$ (cf. Equation 5.5). I.e., both levels result in quantities which are high enough for the specification of reliable quality measures for evaluation, like averages of criteria defined over all coalitions which are formed in a single run.
Next, factor X_8 determines the percentage of agents which starts the process of coalition formation as an initiator. Taking into account the possible total numbers of agents as defined by factor X_7, a low level of 0.1 % and a high level of 0.4 % are equivalent to the absolute amounts of 100 and 400 initiators if $|A| = 1000$, and 200 and 800 initiators if $|A| = 2000$ (cf. Equation 4.22). In this regard, recall from Section 4.3.2.4 that in the

$\|A\|$	$e^{(i)}_{p_{tar}}$	$\|C^*\|$	$\|CS^*\|$
1000	100	10	100
1000	200	20	50
2000	100	10	200
2000	200	20	100

Table 5.7: Sizes of optimal coalition structures.

course of coalition formation responders switch roles if specific conditions are met, so these values only refer to the initial state of the process.
With regard to the activity of neighborhood formation, factor X_9 defines the extension count ext_N by which initiators expand their neighborhood if coalition formation with current neighbors failed. Considering the optimal coalition sizes of 10 and 20 members, a low level of 10 agents equals the full and the half size of an optimal coalition, respectively. In contrast, a high level of 40 agents considerably exceeds the optimal sizes but takes account of the fact that initiators potentially scan large parts of the grid in their search for new cooperation partners. The number of consecutive unsuccessful initiations before an agent extends its neighborhood is defined by factor X_{10}. While a low level of 2 initiation attempts is intended to restrict the communication cost of the formation process by avoiding unnecessary retries, a high level of 8 attempts aims at minimizing the scope of the resulting coalitions by raising the chance to cooperate with nearer neighbors.
With respect to the interaction protocol which is used by agents for the formation of coalitions, factor X_{11} determines the maximum number of proposals which an initiator accepts at once (cf. Algorithm 4.3). While in case of the low level only the most beneficial reply is agreed to, in case of the high level an agent accepts 10 proposals at the same time.
Next, factor X_{12} specifies the time t_{bi} after which a responder a switches its interaction role with a defined probability $P(bi)$ because of a lack of received calls for proposals. Assuming that $a \in C$, we generally determine for all simulations that $P(bi) = 1$ if $|C| > 1$, and $P(bi) = r_{init}$ otherwise. I.e., a responder switches roles with a probability of 100 % if it is representative of a non-singleton coalition and with a probability equal to the initial percentage of initiators if it still forms a coalition on its own. With regard to the chosen levels of t_{bi}, recall from Section 5.1 that in a simulation a single simulation step covers the period of 100 ms. Thus, the defined levels of 2 min and 8 min correspond to the number of 1200 and 4800 simulation steps in which calls can be received, respectively. The maximum number of times which a responder tries to become an initiator is specified by parameter X_{13} which is varied between 2 and 8 attempts.
Finally, factor X_{14} defines the maximum number of optimization cycles oc_{max} which are conducted by agents in the course of coalition formation in order to approximate the values of an already fulfilled target product to a higher degree. While the low level

restricts the activities to a single optimization attempt, the high level allows for a total of 40 optimization cycles.
Having discussed the settings of the different parameters, we are ready to specify the two experiments which are performed in order to answer research questions $RQ3$ and $RQ4$:

*E-RQ*3 Given the parameterization shown in Table 5.6, a series of simulations is conducted based on a two-level full factorial design while the resulting cumulative contributions con_C are captured as responses.

*E-RQ*4 Given the parameterization shown in Table 5.6, a series of simulations is conducted based on a two-level full factorial design while the resulting coalition scopes s_C are captured as responses.

I.e., we examine both research questions by executing a full factorial design and capturing those responses which are relevant for the considered problem. In experiment *E-RQ*3, we first analyze the resulting cumulative contributions in order to investigate the capability of DYCE to approximate a global optimum. More precisely, for a single simulation run we determine the degree of approximation by calculating the corresponding mean coalition value $\overline{v(C)}$ reflecting how well the pursued target products are fulfilled by the coalitions $C \in CS$ on average. Furthermore, we examine the rate at which the target products are successfully fulfilled. I.e., for a given coalition structure CS we assess the share of coalitions which cumulative values $e_C^{(i)}$, $err_C^{(i)}$, and $c_C^{(i)}$ lie within the specified tolerance bands. Assuming that $|CS| = n$, the fulfillment rate fr is accordingly defined as the quotient of the number of successful coalitions $count(C^s)$ and the number of all coalitions in CS, i.e.

$$fr = \frac{count(C^s)}{n}. \tag{5.15}$$

As the mean coalition value does not allow to draw conclusions about the individual attributes of the resulting cumulative contributions, in the course of the following discussion we also provide an overview of the cumulative values of the successful coalitions.
With regard to experiment *E-RQ*4, we then investigate the impacts of neighborhood formation by considering the mean scope of all coalitions formed in a single simulation run. As the coalition scope s_C depends on technical characteristics of the power grid which are varied throughout the simulations, we standardize the value in each run by dividing it by the maximum distance occurring between two units in the given grid. Thus, it holds for all runs that the normalized coalition scope $s_C^n \in [0, 1]$. For a coalition structure CS, with $|CS| = m$, the mean normalized coalition scope is then defined as

$$\overline{s_C^n} = \frac{\sum_{i=1}^{m} s_{C,i}^n}{m}. \tag{5.16}$$

Generally, all of the measures being applied in the context of *E-RQ*3 and *E-RQ*4 (i.e. $\overline{v(C)}$, fr, and $\overline{s_C^n}$) are defined on the interval $[0, 1]$, where a higher value reflects a better quality of the results.

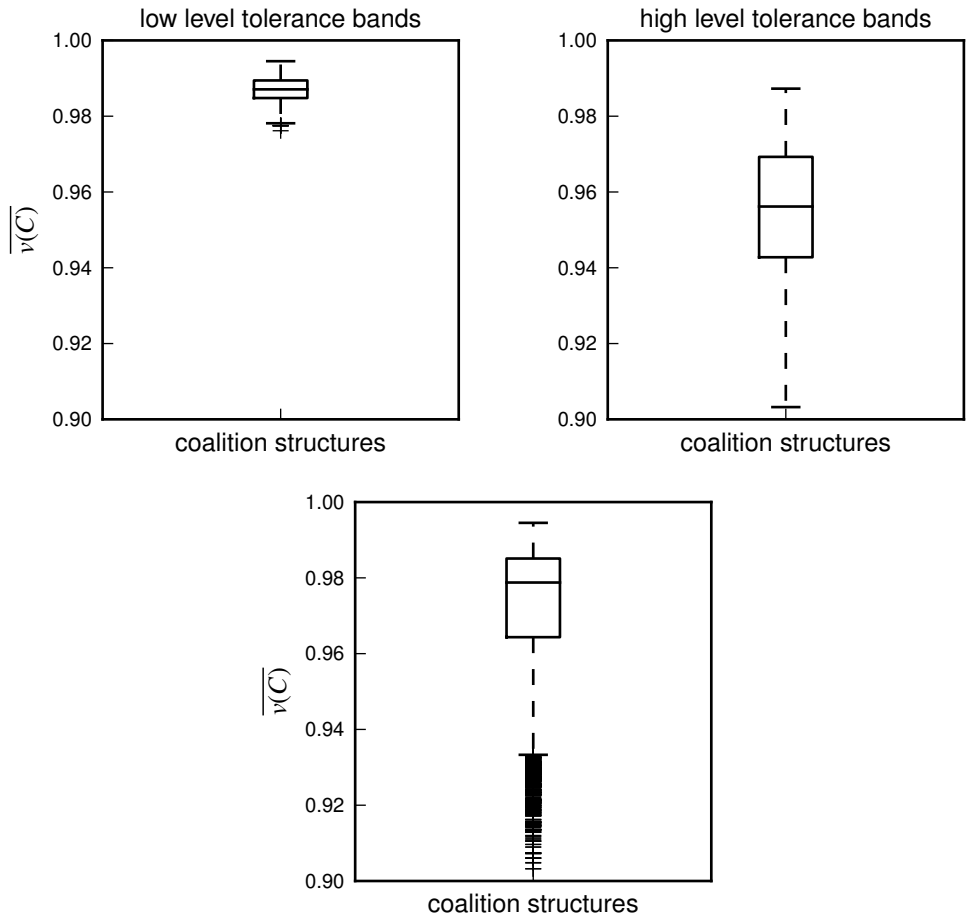


Figure 5.6: Mean coalition values $\overline{v(C)}$.

We start the discussion of the results by considering the mean coalition values $\overline{v(C)}$ which were obtained throughout the simulations of experiment *E-RQ*3 as shown in Figure 5.6. Since a coalition value reflects the degree of approximation to the values of a target product and coalitions continue the formation process until the values of their cumulative contributions lie within the corresponding tolerance bands, the upper left and the upper right plot describe those runs in which all bands were set to the low and the high level, respectively. In both cases, this was true for the number of 2048 simulations. Contrary, the lower boxplot describes the results of all 16 384 simulations. As shown, we generally obtain very good results approximating the global optimum of 1 to a high degree, with a maximum mean coalition value of 0.995. More precisely, for all simulations we retrieve a lower quartile of 0.964, a median of 0.979, and an upper quartile of 0.985. Considering

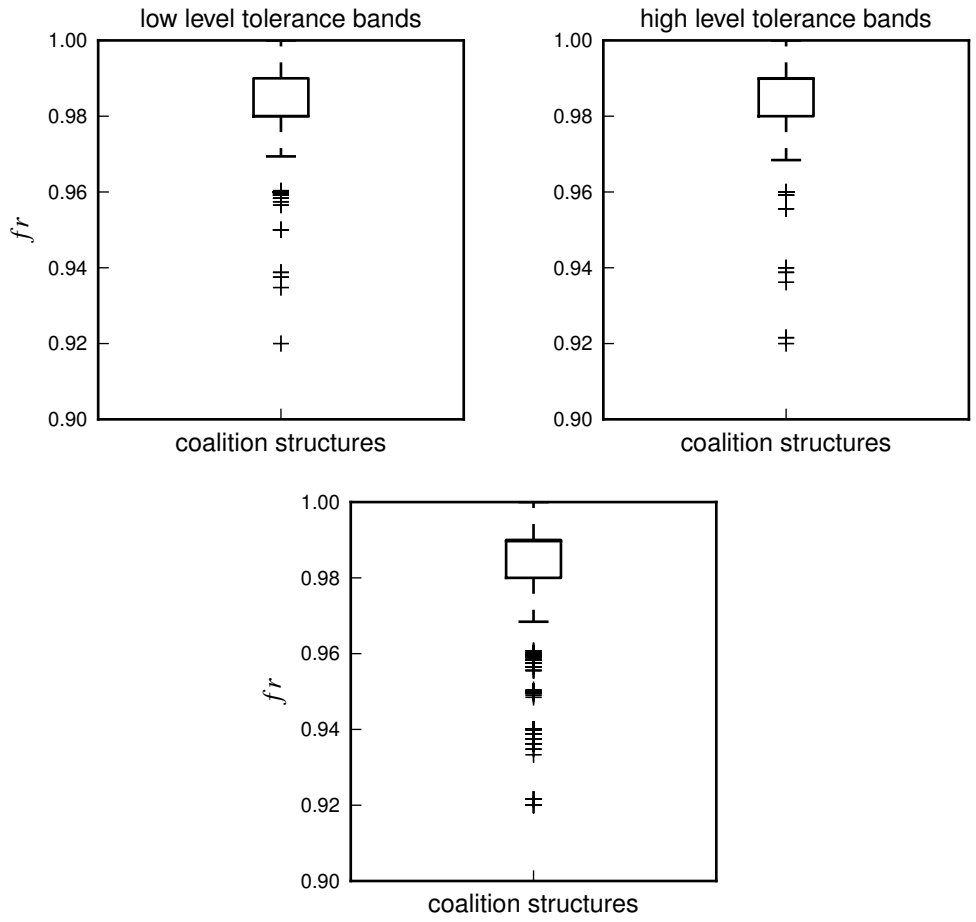


Figure 5.7: Fulfillment rates fr.

that these results reflect the outcomes of all possible factor level combinations (including those being disadvantageous with regard to $\overline{v(C)}$), we can conclude that under the made assumptions DYCE allows for a very good approximation to the global optimum and thus to the values of the pursued target products. As shown by the upper plots, the degree of approximation expectedly improves if the allowed tolerance bands are specified more restrictively. The effects of the different input parameters on the mean coalition value are discussed in more detail in Section 5.3.3.1.

The above results are also reflected by the fulfillment rates fr which were obtained throughout the experiment as depicted by the boxplots in Figure 5.7. Again, the upper plots show the outcomes pertaining to the low level and high level tolerance bands while the lower plot captures the results of all simulation runs. In accordance with the obtained

coalition values, we generally retrieve very high fulfillment rates, with a lower quartile of 0.98, a median of 0.99, an upper quartile of 0.99, and a maximum rate of 1. I.e., in three-quarter of the runs 98 % of the coalitions were able to successfully fulfill their target product. Even in case of the low level bands, which were quite restrictive with regard to the allowed deviations (cf. Table 5.6), we still obtain a median of 0.98, in contrast to a corresponding value of 0.99 in case of the more tolerant high level bands.
With regard to the coalitions which were able to successfully fulfill their target product throughout the experiment, the boxplots in Figure 5.8 show statistics concerning their cumulative contributions. More precisely, the plots on the left capture the mean cumulative electricity amounts $\overline{e^{(i)}_{C^s}}$, mean cumulative errors $\overline{err^{(i)}_{C^s}}$, and mean cumulative costs $\overline{c^{(i)}_{C^s}}$ of the successful coalitions' cumulative contributions $con_{C^s,p_{tar}}$ which were calculated for all simulation runs, i.e. a single mean value pertains to a single run. The plots on the right depict the corresponding standard deviations meaning that a single value refers to the related cumulative values of the successful coalitions' cumulative contributions of a single run. As the target values of the different attributes were varied throughout the experiment, the shown statistics refer to those runs in which the parameters were set to the low level which was true for half of the total number of simulations[3]. As shown by the left plots, the mean cumulative values approximate the pursued target values very well. Moreover, the standard deviations indicate a very low variation among the cumulative values of a single run. For instance, for the mean cumulative electricity amounts $\overline{e^{(i)}_{C^s}}$ we obtain a lower quartile of 99.997 kWh, a median of 100.068 kWh, and an upper quartile of 100.679 kWh, where in three-quarter of the runs the corresponding standard deviation is 3.168 kWh or less. Considering that in half of the cases the corresponding tolerance band was set to the high level of ±8 kWh, these values indicate that the actual deviations from the specified target amount are significantly lower than the allowed tolerance values.
With regard to experiment *E-RQ*4, the boxplots in Figure 5.9 depict the mean normalized coalition scopes $\overline{s^n_C}$ which were obtained throughout the simulations. Since the target electricity amount $e^{(i)}_{p_{tar}}$ of target product p_{tar} determines the number of members of an optimal coalition and thus influences the maximum distance between the included units (cf. Equation 5.4), the left and the right plot show the outcomes pertaining to the low and the high level of the factor, respectively. For the interpretation of the results, recall that in the experiment we did not restrict the coalition scope by an upper threshold, i.e. coalitions were allowed to include units lying arbitrarily far apart. However, the plots in Figure 5.9 show that the final scopes are in fact limited. More precisely, for the target value of 100 kWh and 200 kWh we obtain an upper quartile of 0.44 and 0.51 indicating that in three-quarter of the runs $\overline{s^n_C}$ covers not more than 44 % and 51 % of the grid, respectively. Expectedly, for the high level we get wider scopes because of bigger coalition sizes (on average, the resulting coalitions comprised 10.062 and 20.116 members in case of the low and the high level, respectively). Interestingly, although the

[3] The results for the high levels can be found in Appendix A, Figure A.1. Moreover, Figure A.2 gives an example for the cumulative values of a single simulation run.

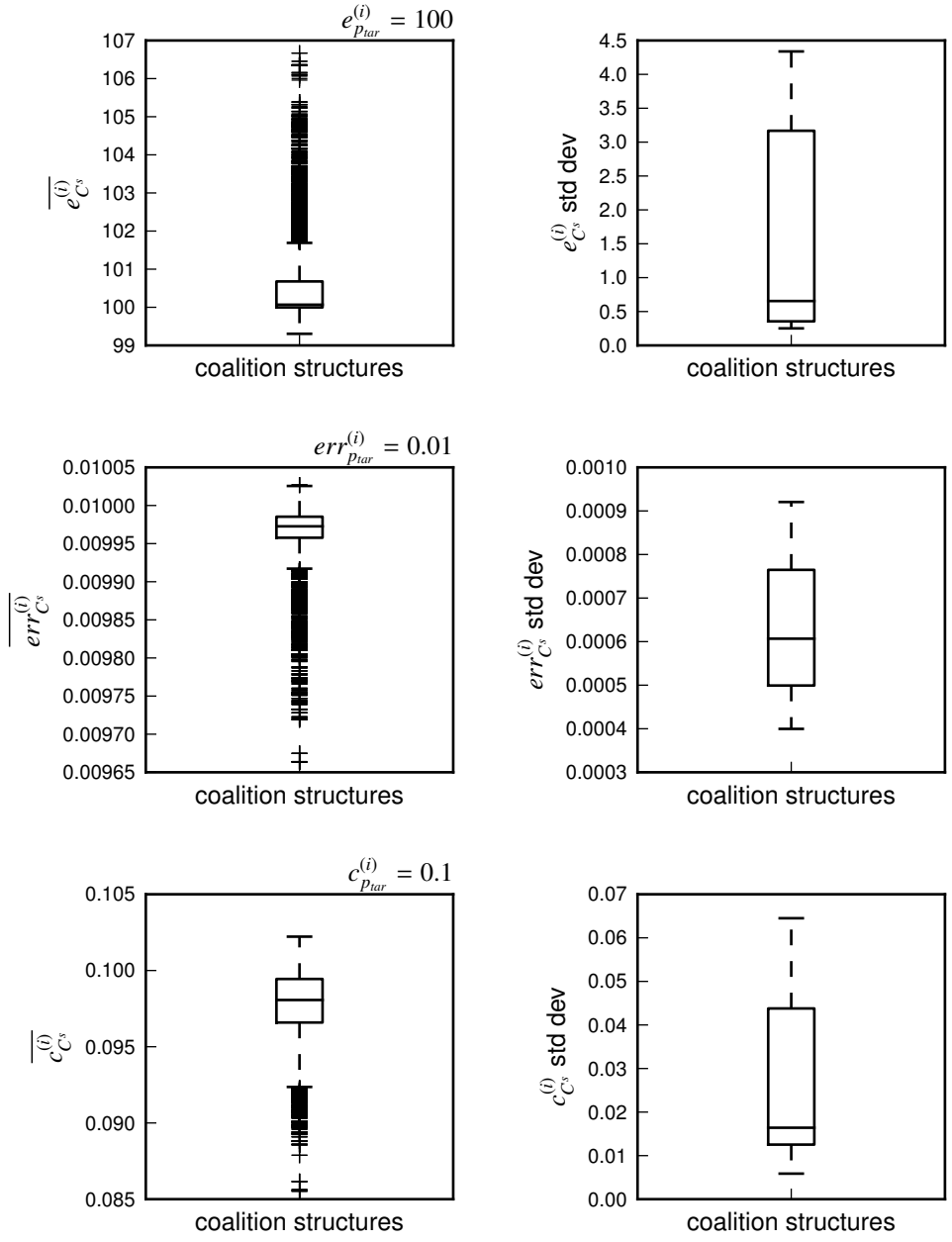


Figure 5.8: Statistics for the cumulative electricity amount, error, and cost.

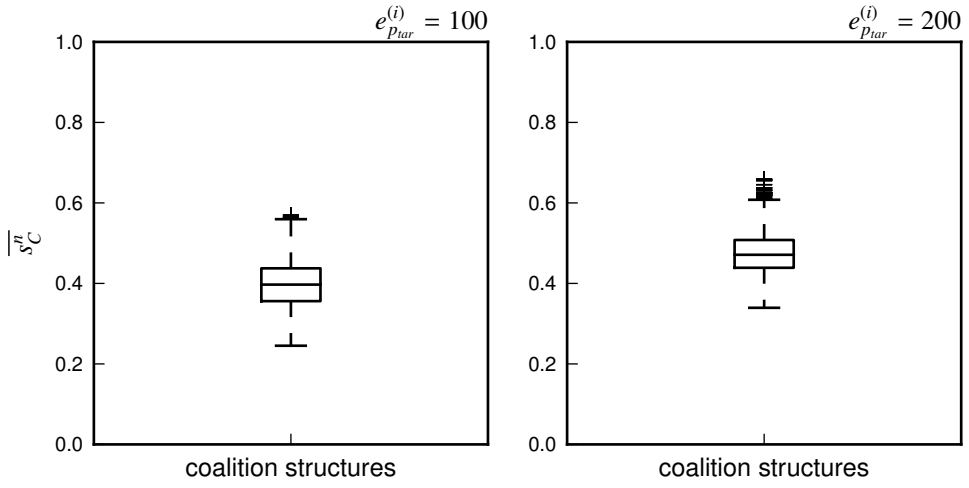


Figure 5.9: Mean normalized coalition scopes $\overline{s^n_C}$.

mean size approximately doubles, the mean normalized scope only increases slightly which suggests a non-linear relationship. All in all, we can conclude that distance-based neighborhoods are an appropriate means to limit the distribution of units of a coalition and exploit the associated potentials as discussed in Section 1.1. If the coalition scope is to be restricted to a defined section in the grid, localized products have to be used which are appropriately accounted for by interaction protocol 4.12 (see also Definition 3.33).

5.3.3 *E-RQ5*, *E-RQ6*: Factor Screening and Response Surface Modeling

In the following sections we provide answers to research questions *RQ*5 and *RQ*6 by first performing a screening of the factors in Table 5.6 for each of the considered evaluation criteria global performance, computational cost, communication cost, and runtime. Based on the results, we then identify the most influential parameters and model the corresponding response surfaces in the form of polynomial functions. This way, we provide a detailed view of the global performance and efficiency of DYCE and reveal the impacts of the relevant input parameters on the considered criteria. The conducted experiments are based on two different types of experimental designs, where a single simulation run is again given by a single execution of DYCE. With regard to the measures which are applied in order to quantify the considered criteria, we assess the global performance by using mean coalition value $\overline{v(C)}$. Contrary, the computational cost is measured through the number of times which the CCOR algorithm is called in the course of coalition formation. As the heuristic performs the calculations for the optimization of the global coalition structure, the corresponding number of function calls fc can be used as reason-

able means to reflect the computational efforts of the agents. The communication cost and runtime are finally measured through the number of messages msg which are sent between agents in the course of coalition formation and the total simulated realtime sr which elapses until termination.
Given these measures, the experiments for answering the two research questions are conducted as follows:

*E-RQ*5 For the criteria global performance, computational cost, communication cost, and runtime, a two-level full factorial design is executed based on the parameterization given in Table 5.6, where for each simulation the resulting cumulative contributions con_C, the number of function calls fc, the number of messages msg, and the simulated realtime sr are captured as responses.

*E-RQ*6 For the criteria global performance, computational cost, communication cost, and runtime, a central composite design is executed based on the parameterizations given in Table 5.8-5.10, where for each simulation the resulting cumulative contributions con_C, the number of function calls fc, the number of messages msg, and the simulated realtime sr are captured as responses.

I.e., in experiment *E-RQ*5 we apply a full factorial design as already used in the previous section in order to assess the effects of the different factors and their interactions on the considered evaluation criteria. In this regard, we restrict our investigations to main effects and two-way interactions as discussed above. The results allow us to identify the most influential factors for each criterion which we then use in experiment *E-RQ*6 as input parameters for creating the corresponding response surface models based on appropriate central composite designs. As we intend to examine possible factor interactions, the models are provided in the form of second-order polynomial functions which goodness of fit is assessed through the coefficient of determination and a comprehensive residual analysis as discussed in Section 2.3.4. In case an improvement of the fit seems worthwhile, we also discuss the results which can be achieved by means of higher-order polynomials[4]. Because in experiment *E-RQ*5 we apply a full factorial design in the context of a deterministic system and thus do not face any randomness in the investigations, the significance of a factor or a factor interaction can not be determined by means of confidence intervals (cf. Section 2.3.3). As general rule, we thus consider those factors and interactions as significant which absolute effect $|eff|$ is at least a third as high as the maximal absolute effect $|eff_{max}|$ on a given response. For response surface modeling, we then use the three most significant factors in order to focus on the most pivotal parameters.
With regard to the central composite designs applied in experiment *E-RQ*6, we generally use an inscribed type in order to preserve the bounds of the original parameterization in Table 5.6. I.e., the α levels of the identified factors are always set to the low and the high level of the original parameterization while the remaining levels (-1, 0, and 1) are

[4] Higher-order polynomial functions allow to increase the degree of approximation to the data. However, this benefit comes at the cost of a higher complexity and the risk to overfit the data.

adjusted accordingly (cf. Section 2.3.2.2). The settings of all other factors which are not included in the model are chosen according to the given optimization goal based on the results obtained in experiment *E-RQ*5. More precisely, for each considered criterion we identify that simulation run of *E-RQ*5 which yields the best result in terms of the applied quality measure and parameterize the remaining factors accordingly. If there are several runs yielding the best result, from these we choose one being optimal with regard to the other three criteria. For instance, if there are several runs yielding the maximum achievable global performance, we choose one (or the one) minimizing the computational cost, communication cost, and runtime.
In the following sections, we discuss the results of experiment *E-RQ*5 and *E-RQ*6 structured by the different evaluation criteria. We always start by describing the outcomes of the factor screening and then provide the model of the response surface. As there are 105 main and two-way interaction effects assessable for a criterion, we focus on the 40 highest effects in each case. With regard to the response surface models, we provide line and contour plots for all included parameters and visualize the results of the residual analyses according the descriptions in Section 2.3.4.

5.3.3.1 Global Performance

Factor Screening

The Pareto chart in Figure 5.10 shows the outcomes of experiment *E-RQ*5 for the criterion global performance by depicting the 40 most influential parameters and two-way interactions ordered by their impact on mean coalition value $\overline{v(C)}$. The dashed lines indicate the significance levels being determined by the maximum absolute effect as discussed above. In accordance with the results of experiment *E-RQ*3 which showed a very good approximation to the global optimum, the impacts on mean coalition value $\overline{v(C)}$ are generally low. The highest effect is induced by the tolerance band for the target electricity amount (X_4) which reduces the mean coalition value by -0.023 when increased from the low to the high level. This is reasonable as a wider band allows a higher deviation from the target volume which in turn yields a lower coalition value[5]. The same holds for the cost band (X_6) which has the third highest impact on the response. Still, with a value of -0.01 the effect is not even half as high as in case of the electricity band. This difference can be attributed to the fact that the values of the electricity amount and the error are of different orders of magnitude which has the consequence that changes in the corresponding tolerance bands have different impacts on the coalition value. This interpretation is further supported by the error band which only induces a minor reduction of -0.0002 and is thus not shown in Figure 5.10. At first sight, these results might suggest that the different attributes of a coalition's cumulative contribution also have an unbalanced impact on the coalition value and thus the optimization process in general. However, recall from Section 4.3.2.2 that the coalition value is not calculated on the basis of the cumulative values but the corresponding approximation values which definitions

[5] With regard to the following discussion, recall from Section 2.3.3 that an effect describes the mean impact of a factor averaged over all performed simulations meaning that it has to be interpreted in this sense.

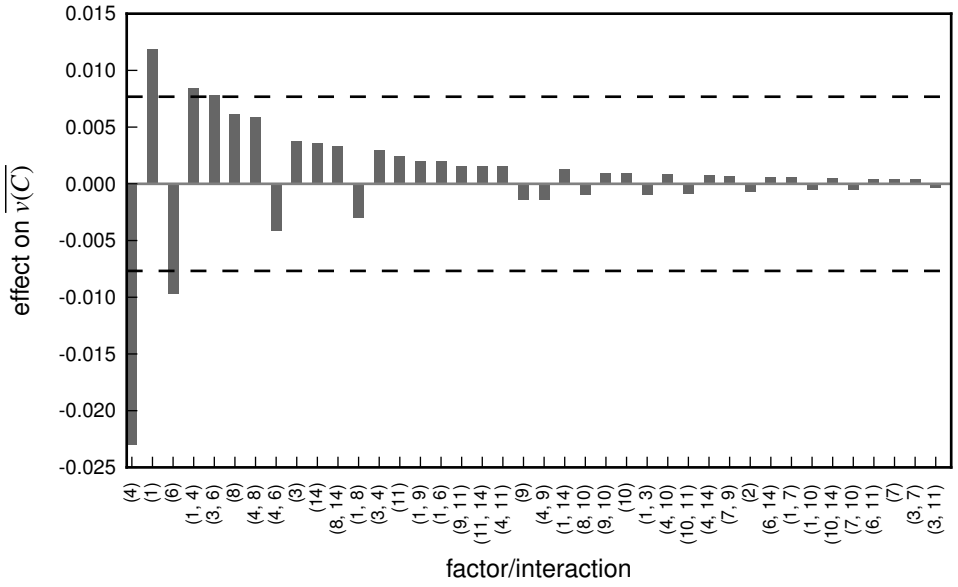


Figure 5.10: Main and interaction effects on the mean coalition value $\overline{v(C)}$.

take account of this problem. As in the course of coalition formation these are further weighed dynamically according the given mean degrees of fulfillment, it is guaranteed that the different attributes are appropriately taken into account.
In contrast to the bands, the target electricity amount (X_1) has a positive effect on the mean coalition value if increased from the low to the high level. This result can be attributed to the fact that in case of a higher target value the size of the optimal coalitions C^* increases while their number in the optimal coalition structure CS^* decreases (cf. Table 5.7). As agents strive for this optimal outcome, eventually less but bigger coalitions form which are able to approximate the values of the pursued target product to a higher degree because of a reduced combinatorial complexity.
Besides the above main effects, Figure 5.10 also shows significant two-way interactions which indicate that the involved factors have to be considered in conjunction. More precisely, the two interactions X_1X_4 and X_3X_6 show that the effect of the target electricity amount (X_1) and target cost (X_3) depend on the level of the respective bands (X_4 and X_6), and vice versa. This is reasonable because a tolerance band specifies an absolute permitted deviation meaning that the relative deviation is bigger if the target value is set to the low level. The fact that the target error does not significantly interact with its band can be attributed to the same reason as discussed above.
All further factors and interactions show only minor effects below the defined significance level. In particular, the maximum number of optimization cycles which are con-

ID	factor	−1	$-\alpha$	0	α	1
X_1	$e^{(i)}_{p_{tar}}$	100	123.274	150	176.726	200
X_4	$e^{(i)}_{p_{tar},-/+}$	0.8	2.476	4.4	6.324	8
X_6	$c^{(i)}_{p_{tar},-/+}$	0.05	0.06	0.08	0.09	0.1

Table 5.8: Parameterization of simulations of experiment *E-RQ*6 (global performance).

ducted by agents in order to further approximate a fulfilled target product (X_{14}) has a positive but rather small effect on the mean coalition value.

Response Surface Modeling

In order to model the response surface of mean coalition value $\overline{v(C)}$, in experiment *E-RQ*6 a central composite design was executed based on the three most significant factors as identified in the context of the previous screening. As shown in Figure 5.10, these are given by the target electricity amount (X_1), the corresponding tolerance band (X_4), as well as the tolerance band for the cost (X_6). The values of the respective factor levels are summarized in Table 5.8, whereas the settings of the remaining eleven parameters are listed in Appendix B, Table B.1. Based on the obtained simulation results, a second-order polynomial model was created which describes the relationship between the response and the included factors. This is given by

$$\begin{aligned}\widehat{\overline{v(C)}} = {} & 0.0016X_1^2 - 0.0029X_4^2 + 0.0019X_6^2 \\ & + 0.0111X_1X_4 + 0.0043X_1X_6 - 0.0055X_4X_6 \\ & + 0.0109X_1 - 0.0193X_4 - 0.009X_6 \\ & + 0.9664.\end{aligned} \tag{5.17}$$

The corresponding coefficient of determination $R^2 = 0.97$ indicates a high goodness of fit. The residual analysis of the model is given in Appendix B, Figure B.1[6]. As depicted, the plots show no extreme patterns. More precisely, the probability plot in the upper left corner indicates that the residuals approximately follow a normal distribution. Furthermore, the following three scatter plots show that the values do not depend on specific levels of the considered factors. Finally, the last plot indicates that the residuals have constant variance as these are randomly distributed about 0.

The above model allows to visualize the response surface of the mean coalition value by means of line and contour plots as shown in Figure 5.11. Each line plot describes the predicted value $\widehat{\overline{v(C)}}$ as a function of one of the considered factors under the assumption that the other two parameters are set to level 0. Correspondingly, each contour plot visualizes the relationship between $\widehat{\overline{v(C)}}$ and a specific two-way interaction.

[6] As discussed in Section 2.3.4, examining the independence of residuals from time makes sense in the context of nondeterministic systems only. Thus, we do not consider this aspect in the following residual analyses.

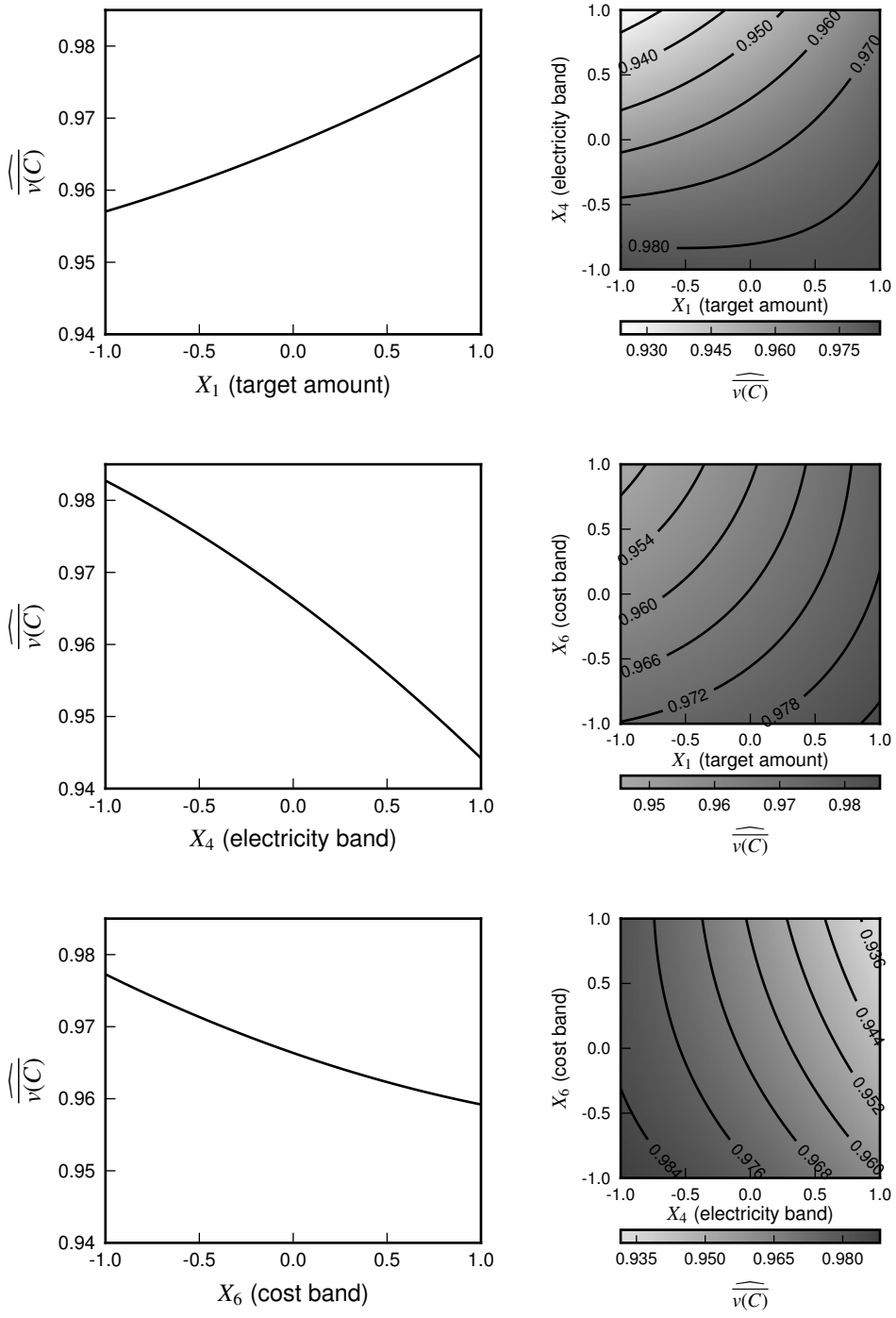


Figure 5.11: Response surface of the mean coalition value.

Taking into account the results of the previous sections, the line plots are generally in line with the mean coalition values and effects obtained in experiment *E-RQ*3 and *E-RQ*5, respectively. More precisely, the first plot shows that the predicted mean coalition value steadily increases with an increasing target electricity amount until reaching a maximum of 0.979. Contrary, the value decreases with a widening electricity and cost band as depicted by the second and third plot. Note that the effects which were obtained in the context of the previous screening are still reflected by the coefficients of the model[7] and the depicted curves. For instance, the second curve shows that the electricity band induces the highest effect among all three criteria. However, we now gain a detailed picture of the factors' impacts between the low and the high level which was previously unobservable. More precisely, the curves show that the mean coalition value increases or decreases almost linearly with an increasing factor level which is a valuable new information with regard to the global performance and appropriate parameterization of DYCE.

The interactions between the three factors are described by the contour plots on the right hand side of Figure 5.11. Besides illustrating the dependency between the involved parameters in detail, each contour plot allows to identify the factor levels at which the response takes on its minimum and maximum with regard to the considered interaction and to assess the relevance of the latter in terms of its impact. In this regard, it generally holds that a strong curvature of the contour lines indicates a high significance. This can for instance be seen when analyzing the plots in Figure 5.11 in view of the results obtained in the context of the previous screening. Here, the most influential interaction effect was induced by the target electricity amount and its tolerance band (X_1X_4). Correspondingly, the first contour plot in Figure 5.11 shows the lines with the strongest curvature. These now also illustrate the previously discussed reason for the interaction: As indicated by the horizontal orientation of the lines near the y-axis, the level of the electricity band is more relevant for the mean coalition value if the target amount is set to -1 (meaning that in this case a small change in the level leads to a relatively big change in the response). If the amount increases, the level of X_4 becomes less important because the relation between the tolerance values and the target amount changes, i.e., the relative tolerated deviation becomes smaller. This can be seen from the more vertical orientation of the lines on the right hand side of the plot. Likewise, the level of the target electricity amount is of greater importance if the electricity band is set to the high level. This is reasonable as in this case the tolerated deviation is generally larger. Expectedly, the maximum mean coalition value is achieved if both factors are set to -1, whereas the minimum occurs if the target amount and the band are set to the low and the high level, respectively.

As indicated by the weaker curvature of the contour lines, the other two interactions are of less importance which could already be observed in the context of the screening. In case of the interaction between the target electricity amount and the cost band, the maximum mean coalition value is obtained if the factors are set to the high and the low

[7] Recall from Equation 2.11 that the effect of a factor or an interaction is twice the value of its coefficient.

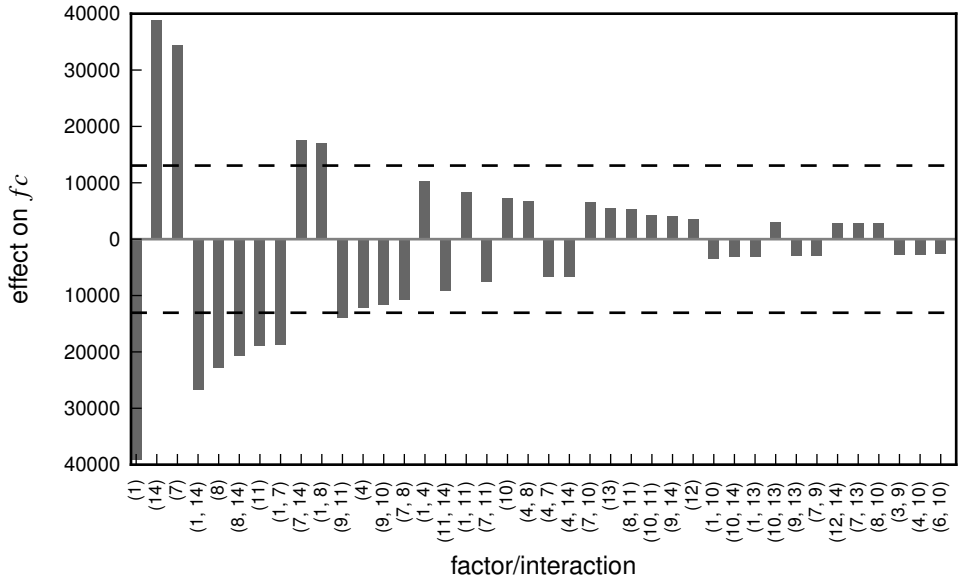


Figure 5.12: Main and interaction effects on the number of function calls fc.

level, respectively. Vice versa, the minimum occurs in just the opposite case. Although the interaction is of less relevance, these results are still reasonable as they reflect the combined effect of the individual factors. Correspondingly, in case of the interaction between both tolerance bands the maximum and minimum occur if both factors are set to the low and the high level, respectively.

5.3.3.2 Computational Cost

Factor Screening

The results of the factor screening for the criterion computational cost are depicted in Figure 5.12. As shown, there are generally more factors and interactions with a significant impact than in case of the global performance. As most influential factor, the target electricity amount (X_1) causes a considerable decrease in the number of function calls when set from the low to the high level. As in this case the size of the optimal coalitions $|C^*|$ and the cardinality of the optimal coalition structure $|CS^*|$ change, the effect reveals that the formation of less but bigger coalitions requires less calculations. A further decrease in the computational cost is induced by an increase of the initiator rate (X_8) which is an interesting insight because a first guess might be that a higher rate leads to an increased number of regrouping attempts and thus to a higher number of function calls. However, a higher rate also accelerates the formation processes which thus require less calculations. Likewise, setting the maximum number of simultaneously accepted

proposals (X_{11}) from the low to the high level also leads to a reduction in the computational cost. This effect can be attributed to the fact that in case of an increase the number of initiations which – from the perspective of the replying agents – finally do not lead to a regrouping is reduced. Assume for instance an initiator which sends a CFP to 40 neighbors from which 20 reply with a proposal. Accepting 10 of these answers and not only the most beneficial one then reduces the number of function calls which were executed by agents which are not part of the final regrouping. In contrast, an increase in the computational cost is induced by a higher maximum number of additional optimization cycles which are conducted by agents when their target product is already fulfilled (X_{14}). While this result can be easily explained by the additional regrouping attempts which are caused by an increase of the parameter, it also shows that the size of the effect has to be taken into account when parameterizing DYCE (see also Chapter 5.3.5). Finally, the last significant factor is given by the number of participating agents (X_7) which increases the computational cost to a similar extent if set from the low to the high level. This effect can simply be attributed to the parameter's impact on the cardinality of the optimal coalition structure and thus the number of coalitions which eventually form.

Besides these main effects, there are also a number of two-way interactions which indicate that the involved factors have a mutual impact on each other and thus have to be considered in conjunction. More precisely, the effect of the target electricity amount (X_1) depends on the maximum number of optimization cycles (X_{14}), the number of agents (X_7), as well as the rate of initiators (X_8), and vice versa. These interactions can be explained by X_1's impact on the cardinality of the optimal coalition structure. As the latter is also influenced by the number of agents, both parameters have a mutually dependent influence on the computational cost as revealed by interaction X_1X_7. Moreover, if the target amount is set to the high level and thus less coalitions form, less optimization cycles are performed as well (X_1X_{14}). In this case, an increased rate of initiators also has a lower reducing impact on the computational cost because less formation processes are carried out in total, whereas the reducing effect of the target amount is lower if the initiator rate is high (X_1X_8).

As further interactions, the effect of the maximum number of optimization cycles also depends on the rate of initiators as well as the number of participating agents, and vice versa. The first interaction (X_8X_{14}) might not seem obvious at first sight because the optimization cycles are executed at the end of a formation process and the initiator rate does not influence the cardinality of the optimal coalition structure meaning that it has no decisive impact on the number of coalitions which eventually forms. However, a higher initiator rate also results in an increased parallelism, i.e. in the most extreme case coalitions form and finally fulfill their target products simultaneously. From an initiator's point of view, the set of potential cooperation partners for the additional optimization activities is then quite limited which results in a decreased number of function calls. Correspondingly, a lower initiator rate causes coalitions to form in a more sequential fashion in which case optimization-related regrouping requests can be sent to a higher number of addressees. More obviously, the interaction with the number of participating agents (X_7X_{14}) can be explained by the latter's influence on the cardinality of the optimal

ID	factor	-1	$-\alpha$	0	α	1
X_1	$e^{(i)}_{p_{tar}}$	100	123.274	150	176.726	200
X_7	$\|A\|$	1000	1233	1500	1767	2000
X_{14}	oc_{max}	1	10	21	31	40

Table 5.9: Parameterization of simulations of experiment *E-RQ*6 (computational cost & communication cost).

coalition structure which in turn has an impact on the number of executed optimization cycles and thus the computational cost.
Finally, the last significant dependency exists between the extension count by which agents expand their neighborhood and the maximum number of simultaneously accepted proposals (X_9X_{11}). As already discussed above, increasing the latter leads to a reduction in the computational cost because the number of initiations which finally do not lead to a regrouping is decreased. This effect is intensified if the extension count is set to the high level as in bigger neighborhoods calls for proposals are sent to a higher number of neighbors.

Response Surface Modeling

With regard to the computational cost as reflected by the number of function calls fc, the three relevant factors for response surface modeling are given by the target electricity amount (X_1), the number of agents (X_7), as well as the maximum number of optimization cycles (X_{14}). The parameterization which was used for the specification of the executed central composite design is shown in Table 5.9, while the settings of the remaining factors can be found in Appendix B, Table B.2. The model which was created on the basis of the obtained simulation results is given by

$$\begin{aligned}\widehat{fc} = &-289X_1^2 + 693X_7^2 + 3164X_{14}^2 \\ &- 6052X_1X_7 - 10781X_1X_{14} + 8349X_7X_{14} \\ &- 9765X_1 + 8422X_7 + 14894X_{14} \\ &+ 17806.\end{aligned} \tag{5.18}$$

With a value of $R^2 = 0.94$, the corresponding coefficient of determination reveals a high goodness of fit. Furthermore, the plots in Appendix B, Figure B.2, indicate that the residuals of the model follow a normal distribution and have constant variance. Moreover, there are no extreme dependencies on the levels of the included factors.
The line and contour plots which visualize the response surface of the computational cost are shown in Figure 5.13. From the first line plot we can see that the predicted number of function calls decreases almost linearly with an increasing target electricity amount. As the latter has an impact on the number of coalitions which eventually forms, this means that if we steadily increase the coalition size and thereby reduce the cardinality of the final coalition structure, the computational cost is reduced as well. The opposite holds in

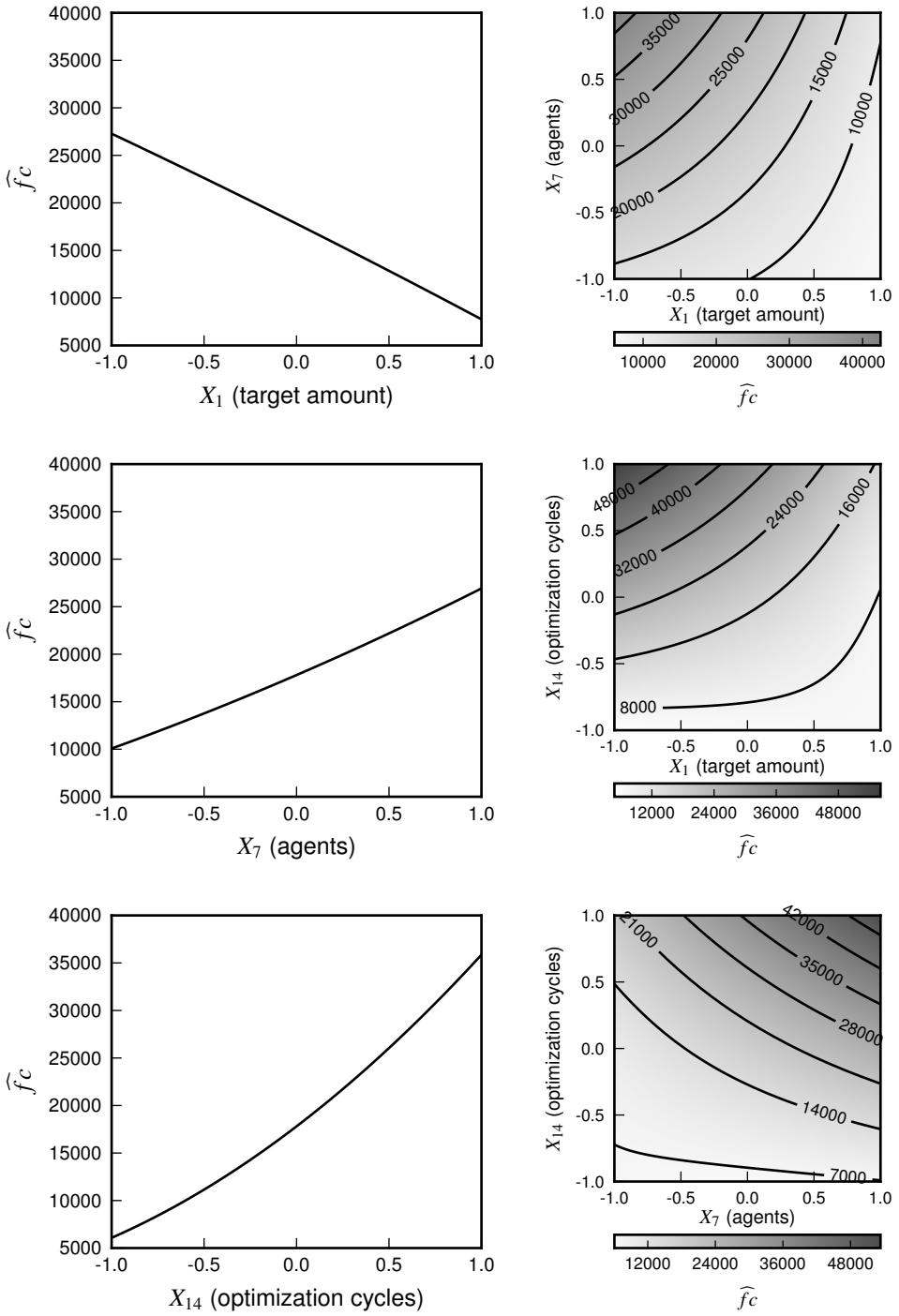


Figure 5.13: Response surface of the number of function calls.

case of the number of participating agents as shown by the curve of the second line plot. Here, a steady increase of the factor causes a nearly linear increase in the computational cost. For a distributed method like DYCE, this is a highly desirable result as it indicates that the approach can be applied to scenarios comprising large numbers of actors. With regard to the maximum number of optimization cycles, we finally obtain a similar curve which is characterized by a steeper slope, however. I.e., in contrast to the results from the screening, the factor now even induces the highest absolute effect among the three parameters. We will discuss this outcome in view of the other evaluation criteria when drawing conclusions from experiment *E-RQ*3-*E-RQ*7 in Section 5.3.5.
The interactions between the three factors are described by the contour plots on the right hand side of Figure 5.13. The most significant one involves the target electricity amount (X_1) and the maximum number of optimization cycles (X_{14}) as indicated by the curvature of the corresponding contour lines. The horizontal orientation of the lines near the y-axis shows that the level of X_{14} is more relevant for the computational cost if the target amount is set to -1. If the amount increases, the setting of X_{14} becomes less important because a lower number of coalitions is formed and thus less optimization activities are conducted in general. Contrary, the level of the target amount has a more decisive influence on the response if the number of optimization cycles is set to 1. This is reasonable as well because the number of formed coalitions is of higher relevance for the computational cost if more optimization activities are performed in general. Consequently, the variance of the response values is low if the number of optimization cycles is set to -1. This is also the reason why the minimum computational cost occurs if the target amount is set to the low instead of the high level as suggested by the descriptions above. Contrary, the maximum cost is expectedly induced if the amount and the optimization cycles are set to -1 and 1, respectively.
As depicted by the first contour plot, the interaction between the target electricity amount and the number of agents has similar characteristics, where the curvature of the contour lines indicates a lower significance. Here, both factors have a more decisive influence on the computational cost if the other parameter is set to a level which leads to a higher number of formed coalitions. For the target amount and the number of agents this holds when the factors are set to the low and the high level, respectively. These are also the settings which cause the highest coalition count in conjunction and thus the maximum computational cost. Contrary, the minimum cost occurs in just the opposite case. Finally, the last contour plot shows the interaction between the number of agents and the maximum number of optimization cycles. Again, each factor has the most decisive influence on the response if the other one is set to a level which induces the highest computational cost. As for both parameters this is the high level, the maximum and minimum cost occur if these are set to 1 and -1, respectively.

5.3.3.3 Communication Cost

Factor Screening

Figure 5.14 shows the results of the factor screening for the criterion communication cost. Generally, the three most relevant factors are the same as in case of the compu-

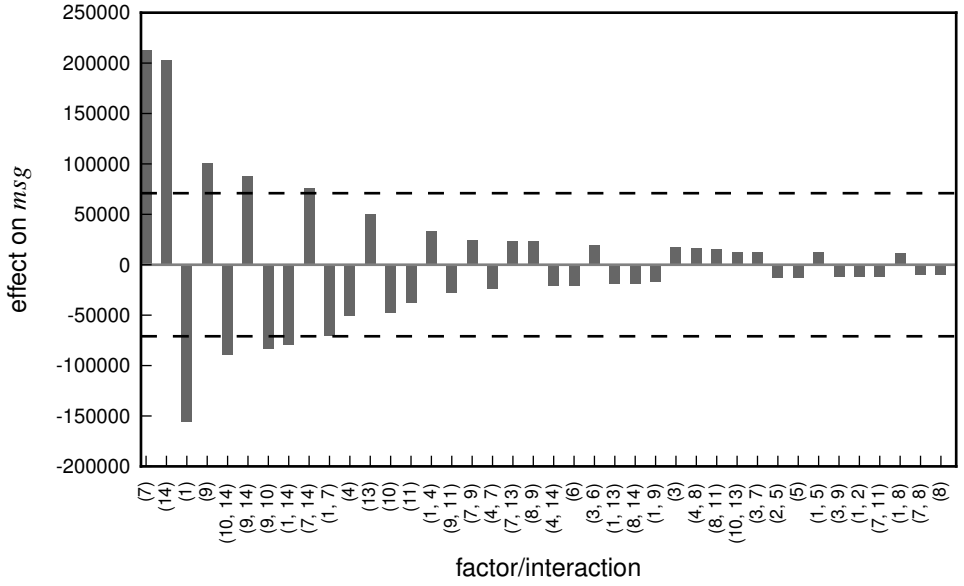


Figure 5.14: Main and interaction effects on the number of messages *msg*.

tational cost, where their relative importance differs as reflected by the order of their effects. The parameter with the biggest impact is given by the number of agents (X_7) which considerably increases the amount of transferred messages if set from the low to the high level. As in case of the computational cost, this effect can be simply explained by the factor's impact on the cardinality of the optimal coalition structure and thus the number of coalitions which eventually forms. Furthermore, increasing the maximum number of optimization cycles (X_{14}) also raises the message count because of the additionally induced regrouping activities. Both effects are of similar size showing that the higher optimization efforts almost cause the same communication load as the doubling of the number of agents. A further significant but smaller increase in the communication cost is induced by a change of the extension count (X_9) which simply results from the fact that in bigger neighborhoods messages are exchanged between a higher number of agents. Interestingly, in the context of the computational cost the factor was only involved in a significant interaction and did not play a relevant role on its own. This fact reveals that the extension count considerably influences the message count but not the number of function calls. Contrary, the only significant reduction in the communication cost is induced by an increase of the target electricity amount (X_1) which can again be attributed to the parameter's impact on the cardinality of the optimal coalition structure. Analogous to the case of the computational cost, the effect shows that the formation of less but bigger coalitions requires less communication.

All further significant effects relate to interactions which mainly involve the above discussed factors. The highest dependency exists between the number of unsuccessful initiations before a neighborhood is extended and the maximum number of optimization cycles ($X_{10}X_{14}$). As described in Section 4.3.2.4, setting the first parameter to a high value is generally intended to prevent a premature neighborhood extension by raising the chance that initiators cooperate with nearer neighbors. In the context of experiment *E-RQ*5, we can see this effect from the factor's diminishing impact on the mean normalized coalition scope (cf. Equation 5.16) as depicted by the Pareto chart in Appendix A, Figure A.3. More precisely, an increase of X_{10} results in a decreased coalition scope which indicates a reduced size of the final neighborhoods. The latter explains the factor's interaction with X_{14} as in case of a smaller neighborhood size the impact of the optimization cycles on the communication cost is limited as well. Because of its influence on the chance of a neighborhood extension, X_{10} also interacts with the extension count which effect on the number of messages is obviously more restricted if neighborhoods are expanded less often (X_9X_{10}). In turn, the extension count further interacts with the maximum number of optimization cycles which can be attributed to the fact that the additional optimization efforts cause a higher communication load if involving more neighbors (X_9X_{14}). The last significant interaction finally indicates a dependency between the target electricity amount and the maximum number of optimization cycles (X_1X_{14}) which can be again explained by the former's impact on the cardinality of the optimal coalition structure. I.e., a higher number of coalitions results in increased optimization efforts which in turn cause a higher amount of transferred messages.

Response Surface Modeling

The previous assessment of the effects on the number of sent messages *msg* showed that the relevant factors for response surface modeling are given by the same parameters as in case of the computational cost, i.e. the target electricity amount (X_1), the number of agents (X_7), and the maximum number of optimization cycles (X_{14}). Thus, the executed central composite design was specified on the basis of the parameterization already shown in Table 5.9, whereas the remaining factors were set as listed in Appendix B, Table B.3. The second-order polynomial which was derived from the simulation output is defined as

$$\begin{aligned}\widehat{msg} = &-5287X_1^2 + 24695X_7^2 + 6448X_{14}^2 \\ &- 6103X_1X_7 - 13522X_1X_{14} + 33962X_7X_{14} \\ &- 30636X_1 + 54837X_7 + 53536X_{14} \\ &+ 99994.\end{aligned} \tag{5.19}$$

The corresponding coefficient of determination $R^2 = 0.92$ and the residual analysis in Appendix B, Figure B.3, indicate that the model fits the data well. In particular, the given plots show that the residuals approximately follow a normal distribution and have constant variance, while there are no extreme dependencies on specific factor levels.
The response surface which is described by the above model is visualized in Figure 5.15. With regard to the individual factors, the depicted curves are generally similar to those

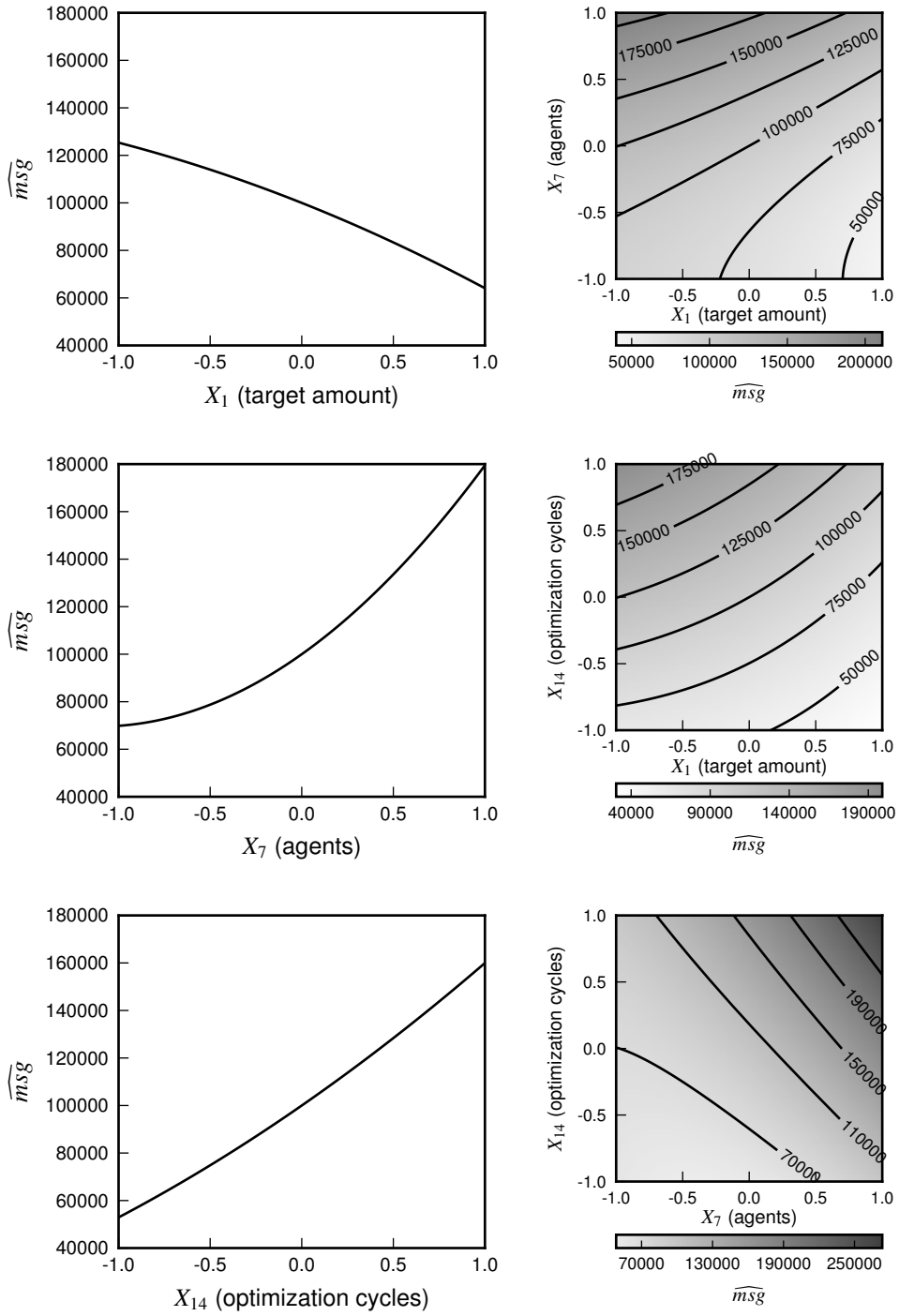


Figure 5.15: Response surface of the number of messages.

obtained in the context of the computational cost, where the most significant difference is given by the effect which is induced by the number of agents in relation to the other two parameters. As shown by the corresponding line plot, in case of this factor the predicted number of messages increases slightly faster than linearly with an increasing factor level. While the induced (absolute) effect is also the highest among all three parameters, it is not significantly bigger than the one caused by the number optimization cycles. Thus, it can be followed that with regard to the communication cost DYCE still scales well with the number of agents. We will further examine this aspect in the context of the last experiment in Section 5.3.4. In contrast, the curves of the other two factors show an almost linear relationship, where an increased number of optimization cycles leads to a higher and an increased target electricity amount leads to a lower communication cost.
Considering the interactions between the three factors on the right hand side of Figure 5.15, we can also observe similarities to the results obtained in the context of the computational cost. In particular, by taking a closer look we can see that the different dependencies can be explained by the same reasons. In the context of the first interaction, a change of the target electricity amount or the number of agents accordingly has a bigger impact on the response if the other factor is set to a level which results in a higher number of formed coalitions. This is reasonable because a change then affects a higher communication cost resulting from the increased formation efforts. As in case of the individual factors, the highest number of coalitions is formed if the target amount and the number of agents are set to -1 and 1, where the relevance of a parameter decreases if the other one deviates from the respective level. As can be seen from the contour plot, these are also the settings at which the maximum communication cost is induced in conjunction, whereas the minimum cost is obtained if the factors are set to 1 and -0.98. The second interaction shows a similar characteristic meaning that a change of the target amount has a bigger impact on the communication cost if a higher number of optimization cycles is performed in general. Contrary, the latter factor is of greater importance if the target amount is set to the low level and more coalitions are formed in total. In conjunction, the maximum communication cost is induced if the parameters are set to -1 and 1, whereas the minimum cost occurs in just the opposite case. The last interaction can finally be interpreted analogously because an increased number of agents also leads to a higher number of formed coalitions and thus has a similar effect on the communication cost as a decreased target amount. Correspondingly, each factor is of bigger importance for the response if the other one is set to a level which causes a higher number of messages in general, where the maximum and minimum load occurs if both parameters are set to 1 and -1, respectively.

5.3.3.4 Runtime

Factor Screening

As can be seen from the Pareto Chart in Figure 5.16, among all evaluation criteria the runtime is exposed to the highest number of significant effects. The biggest impact is induced by the extension count (X_9) which considerably reduces the runtime when set from the low to the high level. This effect can be attributed to the fact that the high level

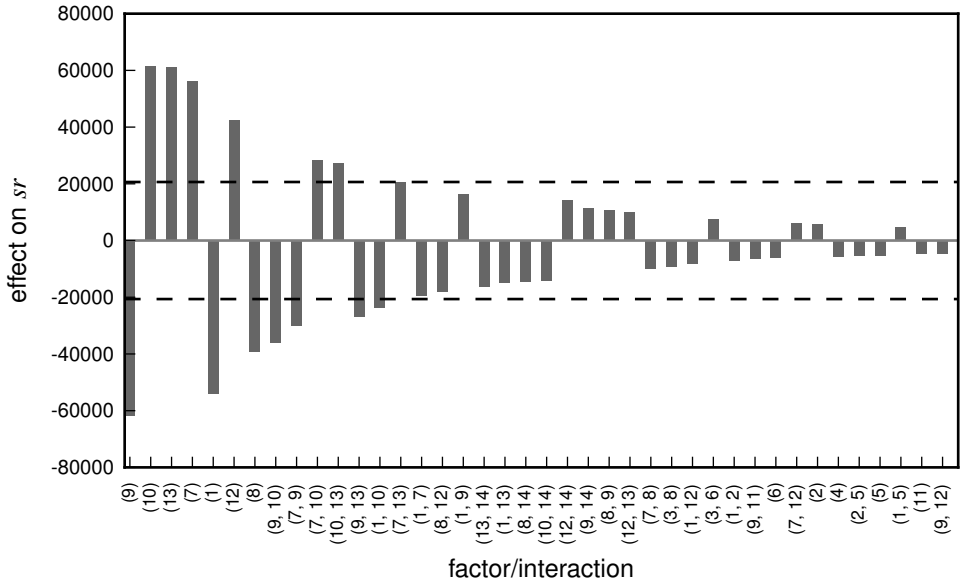


Figure 5.16: Main and interaction effects on the simulated realtime *sr*.

allows initiators to take an increased number of neighbors for their regrouping activities into account and thus to find appropriate cooperation partners faster. However, as shown by the results of the previous section, this benefit generally comes at the cost of a higher communication load.

Further significant reductions in the runtime are caused by the target electricity amount (X_1) and the rate of initiators (X_8). These results are in line with the factors' diminishing effects observed in the context of the computational and communication cost and can be analogously explained by the induced decrease in the number of formed coalitions and the higher degree of parallelism with regard to the conducted formation activities. In contrast, the runtime is increased through a higher number of unsuccessful initiations before a neighborhood is extended (X_{10}). While in this case the high factor level prolongs the formation process if a neighborhood does not comprise a sufficient amount of appropriate cooperation partners in general, it also provides the advantage to raise the chance of a collaboration with nearer neighbors as discussed in the previous section. A longer runtime is further induced by the maximum number of times which an unsuccessful responder switches roles and become an initiator in order to proactively form a coalition (X_{13}). This result is not surprising as a higher number of role changes prolongs the formation process. Likewise, increasing the number of agents (X_7) also results in a longer runtime which can be on the one hand explained by the associated increase in the grid size and on the other hand by the higher number coalitions which is finally

formed. A larger grid has a delaying impact because the extension count is a constant factor and neighborhoods can be expanded up to the whole network. The last significant effect is finally induced by the time which a responder waits for appropriate regrouping requests until switching roles and becoming an initiator (X_{12}). Clearly, increasing the period prolongs the formation process if the agent is not addressed by other participants. Besides the above main effects, there also exist significant two-way interactions between some of the discussed factors. The first one involves the extension count and the number of unsuccessful initiations before a neighborhood is expanded (X_9X_{10}). This dependency can be explained by the fact that a low extension count requires initiators to expand a neighborhood more often in order to include the same amount of agents as in case of a high factor level. Consequently, the delaying effect of X_{10} is intensified because the number of unsuccessful initiations refers to a specific neighborhood size. Furthermore, the extension count interacts with the number of agents (X_7X_9) which can be attributed to the fact that neighborhoods are generally extendible to the whole grid. I.e., the time-saving effect caused by an increase of X_9 is higher if more agents participate and the grid is of bigger size. Finally, the extension count also interacts with the maximum number of times which a responder becomes an initiator (X_9X_{13}). To clarify, assume that X_{13} is set to the high level and a responder repeatedly changes roles but remains unsuccessful in its attempts to proactively form a coalition. Each time the agent takes on the role of an initiator, it accordingly scans the whole grid in search for new cooperation partners. Consequently, the process proceeds faster if the extension count is set to the high level. Furthermore, the example clarifies why X_{13} also interacts with the number of unsuccessful initiations before a neighborhood is expanded ($X_{10}X_{13}$). If an agent becomes an initiator more often and thus scans the whole grid for a higher number of times, the delaying effect of X_{10} is increased. Similarly, this effect is also increased if the grid is of bigger size because in this case neighborhoods are extended more often as indicated by the interaction between the number of participating agents and the number of unsuccessful initiations (X_7X_{10}). Finally, the effect of X_{10} also depends on the level of the target electricity amount (X_1X_{10}) which can be attributed to the latter's impact on the cardinality of the optimal coalition structure. I.e., if the formation process proceeds faster because a lower number of coalitions is formed, neighborhoods are extended less often which reduces the delaying impact of the number of unsuccessful initiations on the runtime.

Response Surface Modeling

According to the results of the previous screening, the relevant factors for modeling the response surface of the simulated realtime *sr* are given by the extension count (X_9), the number of unsuccessful initiations before a neighborhood is extended (X_{10}), as well as the maximum number of times which a responder changes roles in order to become an initiator (X_{13}). The factor levels which were used for the specification of the executed central composite design are summarized in Table 5.10, whereas the settings of the re-

ID	factor	-1	$-\alpha$	0	α	1
X_9	ext_N	10	17	25	33	40
X_{10}	$i_{u,max}$	2	3	5	7	8
X_{13}	bi_{max}	2	3	5	7	8

Table 5.10: Parameterization of simulations of experiment *E-RQ*6 (runtime).

maining factors are given in Appendix B, Table B.4. The created model which describes the relationship between the response and the above parameters is given by

$$\begin{aligned}\widehat{sr} = {} & 42271X_9^2 - 14071X_{10}^2 - 13707X_{13}^2 \\ & + 3989X_9X_{10} + 1305X_9X_{13} + 1170X_{10}X_{13} \\ & - 15271X_9 + 2190X_{10} + 15158X_{13} \\ & + 32790.\end{aligned} \tag{5.20}$$

The corresponding coefficient of determination $R^2 = 0.78$ indicates a slightly increased residual variability. Moreover, the upper plots in Appendix B, Figure B.4, reveal small deviations from the normal distribution and a noticeable dependency on the levels of the extension count, whereas the remaining plots show no further remarkable patterns.
The response surface of the runtime is visualized in Figure 5.17. Because of the reduced goodness of fit, a third-order polynomial model was created for comparison which describes a surface as depicted in Appendix B, Figure B.5. While the corresponding coefficient of determination $R^2 = 0.94$ is higher, the shapes of the curves are not considerably different from those of the second-order model. Moreover, the residual analysis in Figure B.6 shows strong deviations from the normal distribution and dependencies on the factor levels. Thus, we use the second-order model for the following examinations.
Generally, the line plots in Figure 5.17 reveal non-linear relationships between the considered factors and the response. With regard to the extension count, the related runtime behavior is described by a parabola which opens upward and attains its minimum at a level of 0.17. I.e., after an initial decrease, at higher levels the runtime starts to increase again, where the factor induces the highest effect among all three parameters. Contrary, with regard to the other two factors we obtain opposite runtime behaviors which show their smaller impact as can be seen from the covered ranges of y values. As the third parabola attains its maximum at a relatively high level of 0.56, the runtime primarily increases with an increasing number of role changes. Contrary, in case of the number of unsuccessful initiations the curve already decreases from a level of 0.07 onwards.
These relationships are also reflected by the contour plots on the right hand side of Figure 5.17. As can be seen from the contour lines, the first two interactions generally show a similar characteristic, where in both cases the extension count has a decisive influence on the runtime if varied within the lower and upper range. In-between, the factor plays a less important role, whereas the relevance of the other two parameters increases. The first interaction induces the maximum runtime if the extension count and

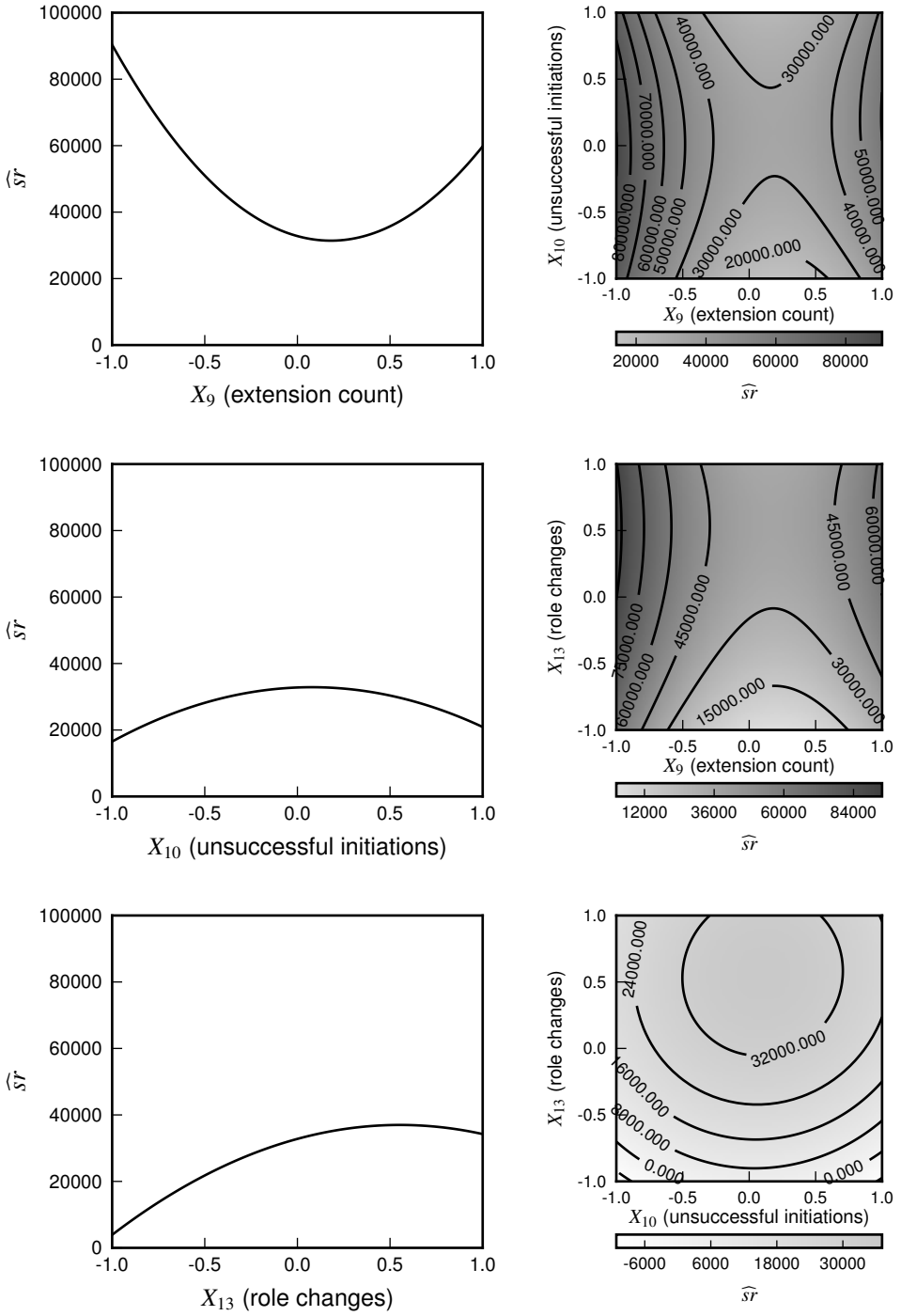


Figure 5.17: Response surface of the simulated realtime.

the number of unsuccessful initiations are set to -1 and -0.07, respectively, whereas the minimum occurs at the levels 0.23 and -1. Note that these results are in line with the curves depicted by the corresponding line plots. Similarly, the second interaction causes the maximum runtime if the extension count and the number of role changes are set to -1 and 0.52, whereas the minimum is obtained at the levels 0.19 and -1. The last contour plot finally visualizes the interaction between the number of unsuccessful initiations and the number of role changes, where the described surface takes on the form of a peak. Correspondingly, the contour lines are arranged concentrically around the maximum which is induced at the levels 0.09 and 0.56. Considering the respective line plots, the latter correspond almost exactly to those levels at which the individual parabolas attain their maximum. Because the surface is formed like a peak, these are also the settings at which a change of one factor causes the biggest change in the runtime if the other one is kept at its level. For the same reason, the relevance of one factor decreases the more, the more the other one deviates from its setting. As a result, the minimum runtime is induced if both parameters are set to -1.

5.3.4 *E-RQ*7: Scalability for Large Numbers of Agents

In what follows, we finally conduct the last step of our evaluation process and provide an answer to research question *RQ*7 by examining the scalability of DYCE with regard to scenarios comprising large numbers of actors. While experiment *E-RQ*5 and *E-RQ*6 already considered this aspect within the bounds of 1000 and 2000 agents, we now complement the results by investigating numbers corresponding to an application on global system level (cf. the use cases discussed at the beginning of Chapter 4). Thus, we assess the scalability of the approach by again using the previously applied evaluation criteria global performance, computational cost, communication cost, and runtime.
The parameterization which is used in the following experiment is summarized in Table 5.11, where a single simulation run is again given by a single execution of DYCE. As shown, the set of factors comprises those already considered in the previous experiments plus an additional parameter $s_{N,max}$ specifying the maximum size to which a neighborhood can be extended. The latter is introduced because larger scenarios put increased requirements on the demanded resources (like computational power or network bandwidth) and a restriction of the neighborhood size provides the potential to handle the very same. However, because a restricted neighborhood size also limits the search space of possible coalition structures in the sense that agents only take a restricted set of participants for coalition formation into account, it may also compromise the global performance of DYCE. In the following experiment, we thus conduct two series of simulations in which we compare the impacts of a limited neighborhood size to those of an unlimited one. Because the considered numbers of agents become very large and neighborhoods may consequently comprise much bigger amounts of participants than in the previous experiments, the levels which were formerly used for the extension count ext_N

		variable		
ID	factor	min	max	fixed
X_7	$\|A\|$	1000	16 000	-
X_9	ext_N	100	1600	-
X_{15}	$s_{N,max}$	1000	∞	-
X_1	$e^{(i)}_{p_{tar}}$	-	-	200
X_2	$err^{(i)}_{p_{tar}}$	-	-	0.01
X_3	$c^{(i)}_{p_{tar}}$	-	-	0.1
X_4	$e^{(i)}_{p_{tar},-/+}$	-	-	0.8
X_5	$err^{(i)}_{p_{tar},-/+}$	-	-	0.01
X_6	$c^{(i)}_{p_{tar},-/+}$	-	-	0.05
X_8	r_{init}	-	-	0.4
X_{10}	$i_{u,max}$	-	-	2
X_{11}	$rCom$	-	-	10
X_{12}	t_{bi}	-	-	2
X_{13}	bi_{max}	-	-	2
X_{14}	oc_{max}	-	-	1

Table 5.11: Parameterization of simulations of experiment *E-RQ*7.

are not reasonable any more. In the conducted simulations, we thus take a new approach and calculate the value dynamically as

$$ext_N = 0.1 \cdot |A|. \tag{5.21}$$

I.e., in each run the extension count is equal to one-tenth of the number of participating agents. With regard to the remaining factors, the levels are generally kept fixed, where the settings in Table 5.11 were determined based on the results of experiment *E-RQ*5. More precisely, as the factors partially had conflicting effects on the evaluation criteria in terms of the goal to increase both the global performance and the efficiency, we gave priority to the global performance by setting all factors which had a significant impact on the mean coalition value to those levels which finally resulted in a higher response value. For instance, because a widening of the electricity and cost band had a negative effect on $\overline{v(C)}$, we chose the low levels for the parameterization. Contrary, the remaining factors were set in favor of the efficiency-related criteria, where the only conflict arose in case of the number of unsuccessful initiations (X_{10}) which previously induced an increase in the computational cost and runtime but a decrease in the communication cost. However, as the effect on the runtime was the only significant one, we chose the low level for the parameterization. Given the latter, the final experiment of our evaluation is carried out as follows:

*E-RQ*7 For each of the maximum neighborhood sizes $s_{N,max} = \infty$ and $s_{N,max} = 1000$, a series of simulations is conducted in which the number of agents $|A|$ is increased in steps of 1000 and the resulting cumulative contributions con_C, the number of function calls fc, the number of messages msg, and the simulated realtime sr are captured as responses.

In each of the two series, we thus consider a specific maximum neighborhood size and steadily increase the number of participants throughout the simulations, where the extension count is determined dynamically according to the above formula. In order to quantify the considered evaluation criteria, we use the measures already applied in the previous experiments, i.e. mean coalition value $\overline{v(C)}$, the number of function calls fc, the number of messages msg, and the simulated realtime sr. The maximum number of agents which is considered in each series is generally bounded by the resources provided by the simulation system. I.e., as in case of an unrestricted neighborhood size each agent has to keep a model of the whole power grid in memory, in the first series the number is limited to 8000 actors, whereas in the second series the double amount can be examined. The results of experiment *E-RQ*7 are shown in Figure 5.18. As can be seen from the maximum neighborhood sizes specified in the legends, the three plots depict the data of both conducted series, where the left axes refer to the efficiency-related outcomes, while the right axis of the upper and middle plot describe the mean coalition values obtained in the first and second series, respectively. In order to facilitate the analysis, we fit linear polynomial functions to the efficiency-related data of the second series (i.e. the black dots) as visualized by the gray regression lines. Before we start the discussion, note, however, that the slopes of the lines are not comparable as the scale of the second left axis (msg) is determined by the maximum value of the first series. Vice versa, the scales of the first and third axis (fc and sr) are determined by the data of the second series which has to be taken into account accordingly.

As depicted by the upper plot, in case of the first series in which the maximum neighborhood size was unrestricted, we obtain an excellent global performance for all executed simulations. For instance, for the largest number of 8000 agents the mean coalition value still equals 0.99. With regard to the efficiency-related criteria, the simulated realtime increases approximately linearly with an increasing number of agents, whereas the number of function calls and the number of messages grow faster. While the examined parameterization thus leads to an efficient runtime behavior, the results also suggest that an unrestricted neighborhood size causes unmanageable computational and communication cost in scenarios comprising ten thousands of agents. This can simply be attributed to the fact that a formation activity can potentially involve all agents in the grid.

However, taking a look at the results of the second series, we can observe that the growth of the computational and communication cost can be successfully restricted by limiting the maximum neighborhood size to an upper threshold. As shown by the second plot, in this case we still obtain a global performance similar to that of the first series, with a mean coalition value of 0.983 for the largest number of 16 000 agents. However, the efficiency-related criteria now only grow approximately linearly with an increasing

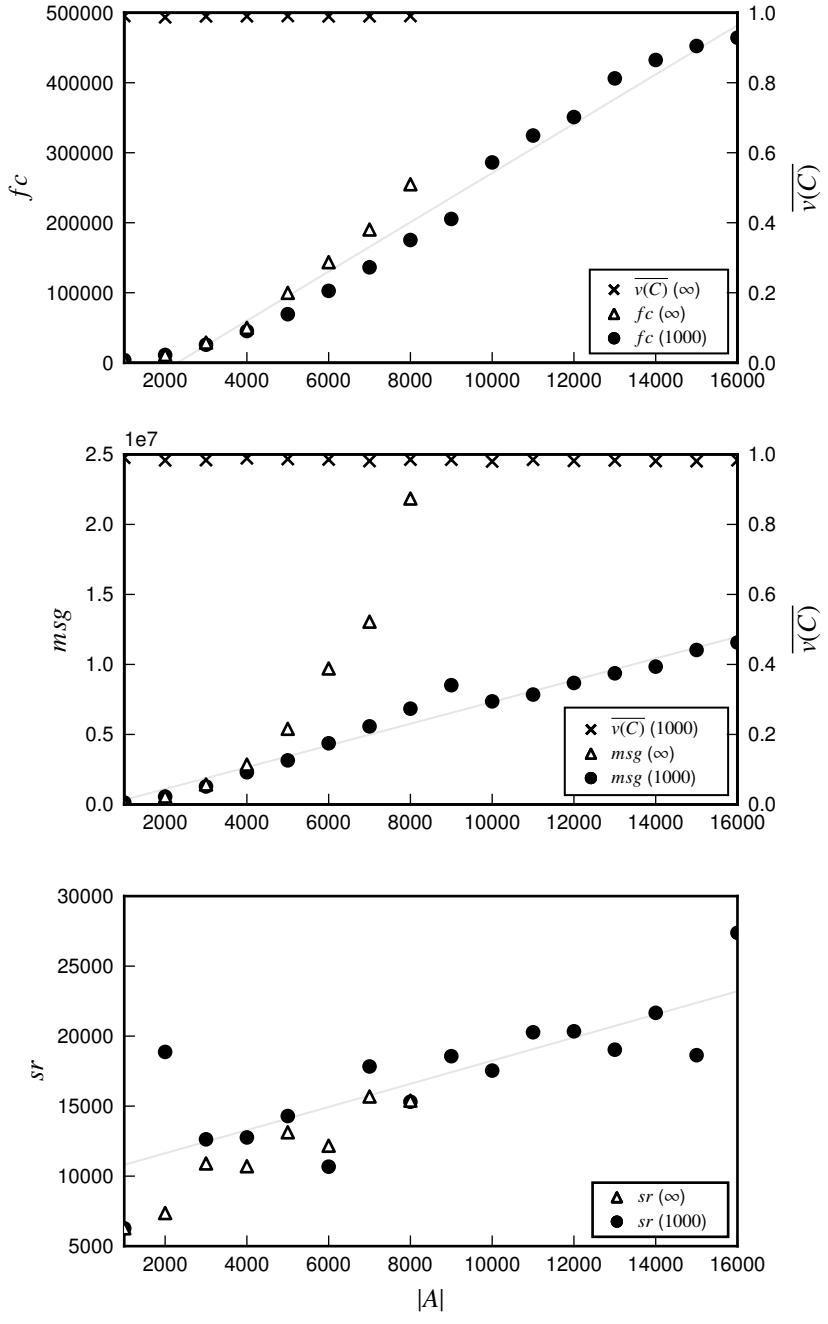


Figure 5.18: Performance- and efficiency-related scalability of DYCE.

number of agents as emphasized by the regression lines. This is a highly desirable result as it shows that DYCE can be effectively and efficiently applied on global system level as well. Still, a restriction of the neighborhood size is only required if the number of agents is very large, whereas in case of amounts up 5000 participants an unrestricted size is totally acceptable.

5.3.5 Summary and Discussion

In the following, we sum up experiment *E-RQ*3-*E-RQ*7 in which we investigated the global performance and efficiency of DYCE making use of concepts from the domain of Design of Experiments. The latter allowed us to handle the high number of input parameters and analyze their impacts on the considered criteria in detail. We initially specified our expectations with regard to the global performance of our approach by defining Hypothesis 1.2 which was formulated as follows:

A distributed and temporally flexible aggregation method enables a high rate of actors to fulfill their power products and allows to approximate the optimal solution of the corresponding global optimization problem to a high degree.

Furthermore, we expressed our assumptions with regard to the efficiency of the method in the form of Hypothesis 1.3 as follows:

A distributed aggregation method is efficiently applicable in terms of the computational cost, communication cost, and runtime which it requires to provide a solution.

We validated the above hypotheses by conducting experiment *E-RQ*3 in which we examined the capability of DYCE to approximate a global optimum and additionally determined the rate at which agents are able to successfully fulfill their pursued target products. In order to allow for a detailed analysis, we chose a two-level full factorial design for experimentation which specified simulations for all 16 384 combinations of the 14 considered input parameters. The degree of approximation and the rate of fulfilled products were measured through mean coalition value $\overline{v(C)}$ and fulfillment rate fr, both being defined on the interval $[0, 1]$. As outcome, we finally retrieved a maximum mean coalition value of 0.996, where the lower and upper quartile of all runs was given by a value of 0.975 and 0.99, respectively. These high values were also reflected by the fulfillment rates for which we obtained a lower and upper quartile of 0.98 and 0.99, respectively, the maximum rate being 1. By taking a closer look at the cumulative contributions of the resulting coalitions, we could finally see that the cumulative values were consistent with the retrieved figures as they showed a high approximation to the values of the pursued target products.
In experiment *E-RQ*4, we then evaluated the effectiveness of the distance-based formation of neighborhoods using again a two-level full factorial design. For these investigations, we applied the mean normalized coalition scope $\overline{s_C^n}$ in order to quantify the average maximum distance which occurs between the units of the coalitions of a final

coalition structure. Although we did not limit the coalition scope in the simulations, the obtained results showed that the final scopes are indeed restricted. More precisely, in the simulations in which the optimal coalition size was given by 10 and 20 members, in three-quarter of the runs the mean coalition scope did not cover more than 44 % and 51 % of the grid, respectively. The results thus showed that the proposed approach for neighborhood formation is suitable for effectively limiting the maximum distance between the units of a coalition which allows to exploit related potentials as initially discussed in Section 1.1.

In experiment *E-RQ*5-*E-RQ*7, we continued to analyze the global performance of DYCE and additionally examined the efficiency-related criteria computational cost, communication cost, and runtime. While the global performance was again measured using mean coalition value $\overline{v(C)}$, the efficiency-related criteria were quantified through the number of executed function calls of the CCOR algorithm fc, the number of sent messages msg, and the simulated realtime sr. In experiment *E-RQ*5, we first determined the effects of the 14 examined input parameters and their two-way interactions on the four criteria based on a full-factorial design as already applied in the previous two experiments. Because the significance of an effect could not be assessed through statistical means due to the reasons discussed in Section 2.3.3, we considered all those factors and interactions as important which absolute effect $|eff|$ was a third as high as the maximal absolute effect $|eff_{max}|$ induced on a given response. Figure 5.19 gives an overview of the experimental outcome by showing the summed effects of all parameters and significant interactions which were determined by first standardizing the absolute effects with regard to the individual responses and then adding up the results over all four of them. A standardized absolute effect thus takes on a value between 0 and 1, whereas a summed effect takes on a value between 0 and 4. Furthermore, Table 5.12 summarizes the signs of all main effects and lists the parameters with wich each factor significantly interacts ordered by the size of the respective interaction effects. The significance of a main effect is indicated by a circle around the corresponding sign. Taking into account both overviews, we can finally draw the following conclusions with respect to the examined factors:

X_1, X_2, X_3 Regarding the attributes of the target product, the target electricity amount (X_1) can be considered as one of the most pivotal factors among the 14 parameters. It not only induces the highest summed effect in total, but also has a significant impact on all four criteria. This relevance can be attributed to the factor's impact on the coalition size and the cardinality of the coalition structure. I.e., if the target amount is increased and consequently less but bigger coalitions form, both the global performance and the efficiency-related criteria are improved. The experimental results also revealed that the factor significantly interacts with a range of other parameters, where all of the considered criteria are concerned. As can be seen from its summed effect, the interaction with the number of optimization cycles (X_{14}) is of particular relevance as it affects both the computational and the communication cost. Taking into account all possible side effects, we can finally conclude that it is generally preferable to set the target electricity amount to the high level.

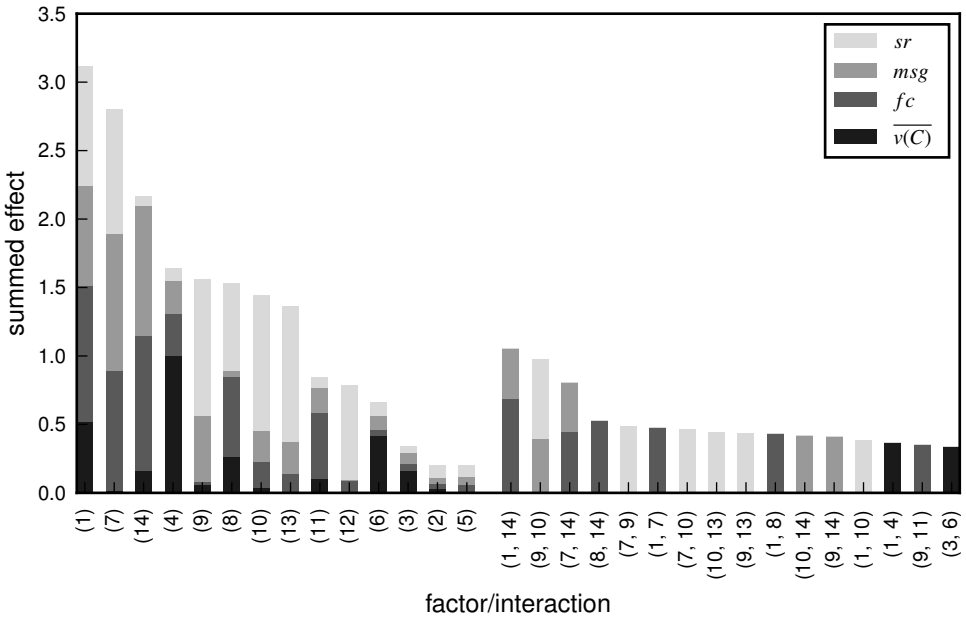


Figure 5.19: Summed effects of all factors and significant interactions.

		effect sign				interacting factors			
ID	factor	$\overline{v(C)}$	fc	msg	sr	$\overline{v(C)}$	fc	msg	sr
X_1	$e^{(i)}_{p_{tar}}$	$\oplus$	$\ominus$	$\ominus$	$\ominus$	X_4	X_{14}, X_7, X_8	X_{14}	X_{10}
X_2	$err^{(i)}_{p_{tar}}$	$-$	$+$	$+$	$+$	-	-	-	-
X_3	$c^{(i)}_{p_{tar}}$	$+$	$+$	$+$	$+$	X_6	-	-	-
X_4	$e^{(i)}_{p_{tar},-/+}$	$\ominus$	$-$	$-$	$-$	X_1	-	-	-
X_5	$err^{(i)}_{p_{tar},-/+}$	$-$	$-$	$-$	$-$	-	-	-	-
X_6	$c^{(i)}_{p_{tar},-/+}$	$\ominus$	$-$	$-$	$-$	X_3	-	-	-
X_7	$\lvert A\rvert$	$+$	$\oplus$	$\oplus$	$\oplus$	-	X_1, X_{14}	X_{14}	X_9, X_{10}
X_8	r_{init}	$+$	$\ominus$	$-$	$\ominus$	-	X_{14}, X_1	-	-
X_9	ext_N	$-$	$-$	$\oplus$	$\ominus$	-	X_{11}	X_{14}, X_{10}	X_{10}, X_7, X_{13}
X_{10}	$i_{u,max}$	$+$	$+$	$-$	$\oplus$	-	-	X_{14}, X_9	X_9, X_7, X_{13}, X_1
X_{11}	$rCom$	$+$	$\ominus$	$-$	$-$	-	X_9	-	-
X_{12}	t_{bi}	$-$	$+$	$+$	$\oplus$	-	-	-	-
X_{13}	bi_{max}	$+$	$+$	$+$	$\oplus$	-	-	-	X_{10}, X_9
X_{14}	oc_{max}	$+$	$\oplus$	$\oplus$	$+$	-	X_1, X_8, X_7	X_{10}, X_9, X_1, X_7	-

Table 5.12: Factor-related effect signs and interacting parameters.

In contrast, the target error (X_2) and target cost (X_3) only have a negligible influence on the different criteria which can be attributed to the fact that they do not directly influence the size and number of the formed coalitions. However, like the target amount, the cost also significantly interacts with its tolerance band regarding the global performance, a dependency which does not exist in case of the error due to the reasons discussed in Section 5.3.3.1.

X_4, X_5, X_6 The tolerance bands for the target values are generally of different importance which primarily results from their different impact on the mean coalition value. In this regard, the electricity band (X_4) induces the highest summed effect followed by the cost and the error band (X_6, X_5). However, as can be seen from the identical signs, the factors still have a similar impact on the different criteria. I.e., in all cases the widening of the band reduces the global performance but optimizes all efficiency-related criteria. The crucial point to note here is that the significant effects merely affect the mean coalition value and the only relevant interactions are given by the two cases already described in the previous paragraph. Thus, we can follow that all bands can safely be set to the low level without risking negative side effects which allows to constantly improve global performance.

X_7 Expectedly, the number of agents is one of the most important factors among all parameters. It has a significant impact on all efficiency-related criteria and induces the second highest summed effect in total, where a higher agent count causes an increase in the computational and communication cost as well as the runtime. Moreover, the experimental results revealed several significant interactions with other parameters which also affect all efficiency-related criteria. As in case of the target electricity amount, the most important one involves the number of optimization cycles which is again relevant for both the computational and the communication cost.

X_8 Choosing a higher rate of initiators allows to significantly reduce the computational cost and the runtime. While the first result was quite surprising at first sight, it is reasonably explainable in light of the second one as a higher rate results in an accelerated formation process requiring a lower number of function calls. Interestingly, the communication cost is only slightly (but still beneficially) affected by an increase which also holds for the global performance. With regard to the computational cost, the factor further interacts with the number of optimization cycles and the target electricity amount. As in both cases the high factor level is preferable, all in all we can conclude that this setting can be used as default in order to constantly improve the computational cost and the runtime.

X_9, X_{10} With regard to the neighborhood-related factors, a higher extension count (X_9) significantly reduces the runtime but also increases in the communication cost. However, a comparison of the standardized effects suggests that the advantage of the runtime reduction outweighs the disadvantage of the increased message load. With regard to the efficiency-related criteria, the factor interacts with a range of other parameters, where in all cases the high factor level leads to a more beneficial result as

well. We can thus conclude that it is generally advantageous to choose this setting as long as communication-related resources are not scarce. With respect to the number of unsuccessful initiations which are accepted by agents before a neighborhood is extended (X_{10}), the experimental results showed that an increase of the factor level significantly increases the runtime but only has a minor impact on the other criteria. However, because at the high level the factor allows to prevent a premature neighborhood extension (see Section 5.3.3.3), it diminishes the negative effect of X_{14} and X_9 on the communication cost as can be seen from the corresponding interactions. Moreover, recall from the Pareto chart in Figure A.3 that, according to the factor's intended purpose, in this case also the mean coalition scope is reduced. Taking into account the further interactions which occur with regard to the runtime, we can thus conclude that for specific goals it is beneficial to set the factor to the high level, while in all other cases the low level is preferable.

X_{11} Considering the number of proposals which an initiator accepts at once, increasing the factor level results in a significant reduction in the computational cost, whereas the effects on the other criteria lie below the defined significance levels but are still advantageous in terms of their signs. As in the context of the only interaction the high factor level is preferable as well, we can follow that this setting solely provides advantages and can thus be used as default.

X_{12}, X_{13} With regard to the factors relating to the role changes which are performed by unsuccessful responders in order to proactively form a coalition, extending the time until an agent becomes an initiator (X_{12}) induces a significant increase in the runtime but no other important main or interaction effects. We can thus follow that the factor has no relevance for the formation process and can safely be set to the low level. Similarly, increasing the number of times which a responder becomes an initiator (X_{13}) causes a longer runtime but no other significant main effect, where the factor interacts with the two neighborhood-related parameters with regard to the same criterion. As in both cases the low factor level yields a more advantageous result, we can conclude that this setting can again be used as default.

X_{14} Considering the number of optimization cycles which are performed by initiators after a target product has already been fulfilled, a higher factor level results in a considerable increase in the computational and communication cost while the effect on the mean coalition value is measurable but not meaningful in terms of the defined significance levels. Moreover, the factor is involved in the highest number of interactions which also affect the computational and communication cost through corresponding side effects. Taking into account that in all these cases the low factor level is more beneficial, we can follow that this setting can be used as default unless the benefit of an ultimate global optimization outweighs the resulting cost.

The above conclusions show that the optimal setting of a factor does not solely depend on its induced main effects but also on the side effects caused by its interactions with other parameters.

Based on the determined impacts, we were next able to identify the three most pivotal factors for each criterion which we used in experiment *E-RQ*6 in order to model the surface of the corresponding responses. More precisely, based on the identified parameters we first executed a central composite design for each criterion and then used the simulation results in order to create a polynomial model with an appropriate goodness of fit. The latter allowed us to study the relationships between the included parameters and the predicted response in detail. In particular, as the models of the computational and communication cost included the number of agents as relevant factor, it could be shown that in both cases DYCE scales well with an increasing factor level.
In experiment *E-RQ*7, we finally analyzed further scalability-related aspects by applying our approach to scenarios comprising large numbers of actors. To this end, we introduced a new factor which allowed us to limit the maximum size of a neighborhood and thus to handle the higher demand for resources. Through two series of simulations, we then compared the effects of an unrestricted neighborhood size to those of a restricted one while successively increasing the number of agents, where we considered both the global performance and the efficiency of our approach. The experimental outcomes showed that in the unrestricted case particularly the communication cost grows faster than linearly with an increasing agent count while the global performance remains at a constantly high level. As a formation process could here potentially comprise all agents in the grid, this outcome was not surprising, though. More importantly, we also observed that a similar global performance can be achieved if the maximum neighborhood size is restricted to 1000 agents in which case the efficiency-related criteria grow approximately linearly with an increasing number of agents. The experiment thus showed that DYCE can be applied on global system level as well.
With regard to our approach to evaluate the method using concepts from the domain of DOE, we can finally conclude that experiment *E-RQ*3-*E-RQ*7 provided a strong argument for their application in the context of deterministic systems. Besides offering the general advantages of an increased efficiency and a wider inductive basis (cf. Section 2.3), the applied experimental designs allowed us to gain detailed insights into the inner workings of DYCE which could not have been obtained via the OFAT approach. More precisely, in experiment *E-RQ*5 the applied full factorial design enabled us to reveal a range of factor interactions many of which were not obvious. As can be seen from the above discussion, these dependencies are generally crucial to consider in order to draw reliable conclusions with regard to an optimal parameterization. Furthermore, in experiment *E-RQ*6 the applied central composite designs allowed us to create detailed models of the response surfaces and thus to analyze how the considered evaluation criteria depend on the included factors. It can thus be followed that, if adapted to the specifics of deterministic systems, DOE is highly suitable for the examination and assessment of algorithms in general and optimization algorithms in particular.
All in all, we can finally conclude that the comprehensive results of experiments *E-RQ*3-*E-RQ*7 support both Hypothesis 1.2 and Hypothesis 1.3. The outcomes showed that DYCE enables a very high percentage of agents to fulfill their target products and allows to approximate the maximum mean coalition to a very high degree. Furthermore, the

experiments revealed that the approach is efficiently applicable in terms of the computational and communication cost as well as the runtime. With regard to the computational cost, this also holds for the CCOR algorithm itself for which we previously determined a worst case complexity of $O(l \cdot m \cdot n)$. Thus, we can follow that DYCE provides an effective and efficient approach for coalition formation being applicable on both local and global system level.

6 Conclusions

Having provided a comprehensive evaluation of DYCE in the previous chapter, in the following we finally conclude this thesis. In Section 6.1, we first sum up the conducted research as well as the contributions made to the state of the art. Moreover, we reconsider our initially specified requirements in order to discuss the achieved goals. In Section 6.2, we then propose topics for future research which may be addressed in order to detail or extend specific aspects of our developed approach.

6.1 Summary and Discussion

Throughout the previous chapters we proposed DYCE, a decentralized self-organization method for dynamic coalition formation in electricity markets. We motivated, designed, and evaluated the approach in a well-structured research process yielding the contributions of this thesis. These are summarized in Table 6.1 (see also Figure 1.1).
In Chapter 1, we started our work by highlighting the need for a new aggregation method motivated by the shortcomings of present pooling concepts. From the identified deficits, we derived requirements for a new solution which we also used in order to discuss related work from the domains of electric power provision, electronic commerce, and distributed artificial intelligence in detail. Afterwards, we specified our goal to address the prevailing shortcomings by providing an aggregation method which fulfills the assessed requirements. We formulated hypotheses in order to express our expectations with regard to the performance and efficiency of the intended approach and finally concluded the chapter by detailing our applied research methodology and the structure of the following investigations.
In Chapter 2, we next provided an introduction to the fundamental topics making up the background of this thesis. We started by giving an overview of the deregulated electricity industry including the possible market types which can be used in order to achieve an equilibrium between supply and demand. Furthermore, we discussed agents and multiagent systems as fundamental concepts of distributed artificial intelligence and particularly addressed the topics of communication, coordination, and organization. Finally, we gave an introduction to Design of Experiments and pointed out specifics which have to be taken into account in the context of deterministic systems. From the descriptions we concluded that with regard to our intended aggregation method coalitions provide an appropriate organizational structure for the fulfillment of our initially defined objectives.
In Chapter 3, we then laid the basis for the design and evaluation of our new approach by specifying DYCE-FM, a formal model capturing the problem of dynamic coalition formation in electricity markets in detail. Drawing on existing formalisms from the domains of DAI and game theory where appropriate, we defined a total of forty-two interrelated concepts which were organized into four different domains. The given definitions covered all aspects of the considered problem, from the power grid and the connected units, to the assigned agents and their contributions to the formed coalitions, through to the electricity market which allows for the trade of the finally fulfilled target products.

contribution	description
requirements analysis	analysis of the requirements for a new aggregation method based on a discussion of related work
DYCE-FM	formal model being composed of forty two interrelated concepts capturing the problem of dynamic coalition formation in electricity markets in detail
complexity analyses	analyses of the computational complexities associated with the problems of product portfolio management, neighborhood formation, coalition formation, and payoff distribution
DYCE	distributed aggregation method providing solutions to the problems of product portfolio management, neighborhood formation, coalition formation, and payoff distribution including two algorithms for the optimization of local and global value (COPE and CCOR)
runtime analyses	runtime analyses of the COPE and CCOR algorithm based on the O-notation
evaluation study	simulation-based evaluation study which provides detailed insights into the inner workings of DYCE using concepts from the domain of DOE

Table 6.1: Research contributions.

As we intended our approach to be universally applicable, the concepts abstracted from economic and technological specifics while still providing enough detail to fulfill the initially stated objectives.
In Chapter 4, we next provided the main contribution of this thesis by presenting DYCE, a new aggregation method for dynamic coalition formation in electricity markets. The approach was generally conceived to be applicable to energy units of arbitrary type, i.e. producers, consumers, and storage. As design we specified an iterative process which was composed of four main activities being carried out by the controlling agents. Each activity addressed a self-contained subproblem for which we first analyzed the associated computational complexity in order to provide an appropriate solution. Afterwards, we detailed the developed process flow along with the related concepts. The final design of the overall process comprised the activity of

- *product portfolio management*, including an optimization algorithm (COPE) enabling agents to identify a product portfolio and operation schedule optimizing their local utility;

- *neighborhood formation*, including an approach allowing agents to assess the physical distance to other participants based on shortest paths;

- *coalition formation*, including an interaction protocol and optimization algorithm (CCOR) enabling agents to coordinate their actions and form coalitions optimizing global value; and

- *payoff distribution*, including a game-theoretical model allowing agents to distribute a gained payoff in a fair way.

For the COPE and the CCOR algorithm, we further provided an analysis of the runtime which showed that both are efficiently applicable even to large schedule spaces and coalitions, respectively.
In Chapter 5, we finally assessed the performance and efficiency of DYCE through a comprehensive evaluation study. To this end, we formulated appropriate research questions and conducted corresponding simulation experiments based on thoroughly designed scenarios. In this context, we made extensive use of concepts from the domain of DOE which particularly allowed us to study the effects of the considered parameters and their interactions in detail. The experimental outcomes showed that DYCE provides an excellent local and global performance and is efficiently applicable in terms of the associated computational cost, communication cost, and runtime. These results could even be confirmed for large numbers of participating agents.
Having summed up our conducted research, we are finally able to consider the fulfillment of our objectives as specified at the beginning of this thesis in Section 1.3. To this end, we discuss to which extend our proposed aggregation method meets the requirements initially identified in Section 1.1:

R_{DC} *(distributed control)* DYCE fulfills the requirement of a distributed control as all specified activities are executable in a fully decentralized fashion. In particular, the developed interaction protocol allows agents to form coalitions without superordinate control, where deadlocks or livelocks are prevented through effective mechanisms.

R_S *(scalability)* The evaluation study showed that DYCE scales well with the number of agents in terms of the associated computational cost, communication cost, and runtime. This property can be particularly attributed to the integration of the flexibly extendible neighborhoods which allow participants to limit the number of interaction partners to reasonable sizes. As initiators are additionally able to identify unlikely cooperation partners which are removed from a neighborhood before a new CFP is sent, overhead resulting from unsuccessful negotiations is further reduced.

R_{CO} *(combined optimization)* DYCE meets the requirement of a combined optimization as it optimizes the individual utility of the agents in the course of product portfolio management and the value of the global coalition structure throughout coalition formation. The local optimization process is performed on the basis of the COPE algorithm, where the concept of a product template allows to integrate multiple markets of arbitrary type into the decision making process. Contrary, global value is optimized through the CCOR algorithm as well as the additional concepts which were developed in order to promote the formation of successful coalitions. These include

the actual process flow which agents run through in the course of coalition formation as well as the approach which they apply for the assessment of dynamic coalition values. The evaluation study showed that both optimization processes yield high quality results.

R_{TF} *(temporal flexibility)* DYCE fulfills the requirement of a temporally flexible aggregation as it specifies an iterative process allowing agents to form coalitions for the fulfillment of a single product only. In this regard, the activities of product portfolio management and coalition formation are properly integrated in the sense that the local optimization process is rerun if a target product could not be fulfilled.

R_{TC} *(trusted cooperations)* DYCE addresses the requirement of trusted cooperations through the integration of an abstract value which quantifies the trust of an agent in another agent at a specific time. The developed overall process specifies the points at which such values are determined, applied, and updated, where we also outlined an approach for their decentralized assessment based on the transitive trust principle. However, the design of a specific trust model and the examination of possible threat scenarios were left to future work.

R_{PD} *(payoff distribution)* DYCE meets the requirement of a final payoff distribution as it integrates a game-theoretical model which allows for a fair division of a coalition's surplus based on the Shapley value. Although the latter is associated with a high computational complexity, the model's design permits the application of an available heuristic which enables an efficient calculation even for large coalition sizes. Moreover, it can be flexibly extended by further distribution criteria if additional aspects are to be taken into account. While we discussed exemplary Shapley values of a coalition in Section 4.4, a detailed evaluation of the model based on a properly designed market was left to future work.

R_{TA} *(topology awareness)* DYCE meets the requirement of a topology-aware aggregation as the specified interaction protocol allows for the fulfillment of localized target products which refer to precisely defined subsections in the grid. Furthermore, the grid-based distance measure is used by agents in order to form neighborhoods of nearest neighbors in terms of the impedance of the connecting power lines.

The above considerations show that we have fulfilled the initially specified requirements to a high degree. Moreover, as discussed in Section 5.2.4 and 5.3.5, the developed approach meets our expectations with regard to the associated benefits as the results of the conducted experiments support our initially formulated hypotheses. However, throughout our investigations we also encountered some aspects which detailed consideration extended the scope of this thesis and which were thus left as open research topics to future work. Along with further possible extensions to our approach, these are finally discussed next.

6.2 Prospects for Future Work

Our conducted research led to a range of new topics which may be addressed in the context of future investigations. These are as follows:

Concepts for failures in communication As specified by assumption A_1 at the beginning of Chapter 4, in the context of this thesis we supposed that the ICT network which is used by the agents for communication provides perfect transmission quality. However, in practice there may be situations in which messages are transferred with errors or arrive in an undefined order. Thus, appropriate concepts are needed for handling such failures in communication. While this issue generally affects all DYCE activities in the context of which agents interact, particularly the interaction protocol for coalition formation is prone to this kind of errors and thus requires an integration of corresponding mechanisms.

Trust model As specified by assumption A_2 at the beginning of Chapter 4, in the context of this thesis we accounted for the aspect of trust through the use of an abstract value while assuming the specific trust model for its calculation to be given. The design of such a model is thus an open topic which should be addressed by future research. This includes the identification of the trust criteria which are relevant for the problem of coalition formation in electricity markets, like an agent's applied security standards or its behavior in the context of former cooperations. Moreover, a concept has to be conceived which derives a single trust value from these criteria. Finally, although we already outlined how such values can be determined by agents in a decentralized fashion, specific interaction protocols are needed which precisely define the sequence as well as the types of the exchanged messages.

Extensions to product portfolio management As described in Section 4.1.2, in the course of product portfolio management an agent optimizes its expected utility by identifying an appropriate product portfolio and operation schedule based on the COPE algorithm. Throughout the optimization process, it takes the price predictions and costs of the product templates and schedules into account, but does not consider the risks which are associated with the trade of the products. For instance, in bilateral markets there is the chance that counterparties do not meet their obligations, whereas in mediated markets this risk is often assumed by a clearing house which guarantees the final settlement. Future work can address this aspect by integrating concepts for the assessment and handling of trading-related risks. This may also require to extend the COPE algorithm by a corresponding functionality or to develop an approach which takes account for this aspect by adjusting the price predictions of the product templates accordingly.

Trading strategies As described in Section 4.1, an agent determines the cost of a contribution and a target product based on its trading strategy because both values have an impact on its individual payoff. Choosing too high values reduces the chance to successfully form a coalition and find an appropriate trading partner, whereas too low

values decrease the final benefit. Moreover, the trading strategy determines which markets and product templates an agent takes into account when creating a template catalogue in the course of product portfolio management because the different possibilities are typically associated with different payoffs and risks. Given that all these decisions also depend on its past experiences as well as the strategic behavior of the other participants, the development of an optimal trading strategy is a highly complex task. This problem can be addressed by future research based on concepts from domains like microeconomics, game theory, and artificial intelligence as well as work already published in this field.

Alternative notions of distance In Section 4.2.2, we proposed an approach for neighborhood formation based on the distance between agents, where the latter was defined in terms of the impedances of the connecting power lines. Future work may examine alternative notions which either extend the proposed one through the integration of further distance criteria or define completely new meanings. For instance, distance could also be specified in terms of organizational structures in order to allow businesses to restrict the formation of coalitions to units which are operated by themselves or possible subcontractors. An alternative notion could also reflect the preferences of an agent with regard to a cooperation, where more favored participants would be in closer proximity than unfavored ones. Finally, integrating multiple aspects into the distance measure would also be an interesting topic to examine.

Error model In Section 4.3.2.1, we modeled the error with which a unit produces or consumes electrical energy and described how the errors of multiple units are aggregated in the context of a cumulative contribution. In this regard, we assumed that in the context of the considered problem possible correlations between the operational behavior of different units are negligible as coalitions are typically large enough to entail a sufficient technological diversification and geographical distribution. However, future work may study this aspect in more detail by analyzing the composition of coalitions in different scenarios and developing a more sophisticated error model based on the gained insights.

Extensions to the distribution model In Section 4.4.2, we proposed a game-theoretical model which allows for a fair division of a coalition's payoff, where further distribution criteria can be flexibly added in the form of appropriately designed weighted voting games. While in this thesis we considered the electricity amount, error, and cost of a members's final contribution as relevant criteria, future work may examine the integration of further aspects like the trustworthiness of an agent or the time at which it joined a coalition. Furthermore, a more detailed experimental evaluation of the proposed model would be desirable which also allows to identify reasonable quotas for the integrated games. As already mentioned at the beginning of Chapter 5, economic investigations with regard to achievable long-run profits are generally complex because they require to model the trading strategies of the agents as well as the market itself. As final aspect, future work may also examine the applicability of alter-

native distribution schemes, like the game-theoretical concept of the nucleolus which provides for a stable division.

Integration of business processes As described in Section 4.4.2, in the context of the last DYCE activity the members of a successful coalition strive for a trade of their fulfilled product after which they finally distribute the associated payoff among each other. In this regard, we outlined the interactions which take place between the coalition and its trading partner as well as the members themselves. Future work may detail this aspect by providing a specification of the related business processes based on standards like UML or BPMN (Business Process Model and Notation). This would facilitate an automation of the settlement and the transactions which are associated with the distribution process.

A Supplements to Experiment *E-RQ3*

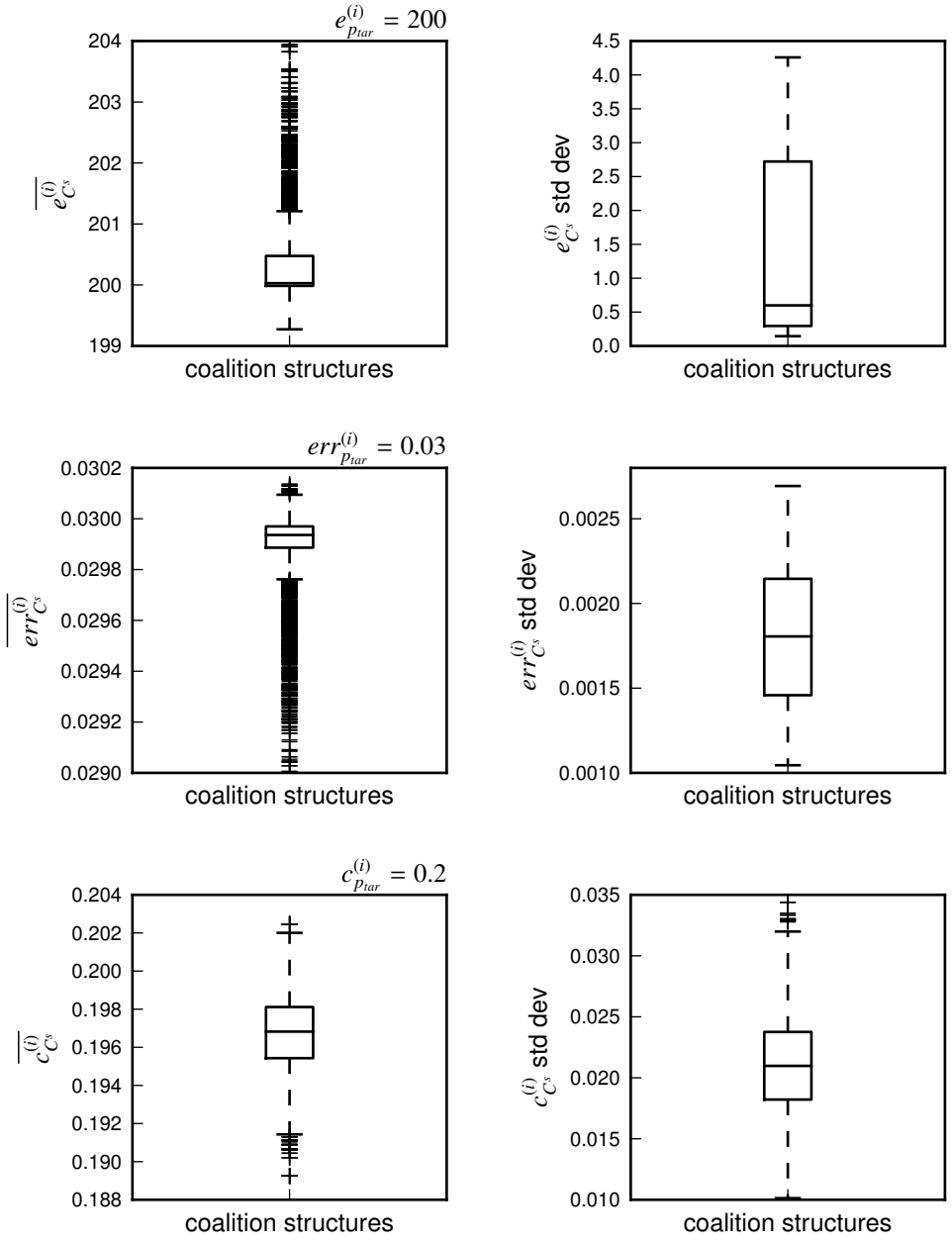


Figure A.1: Statistics for cumulative electricity amount, error, and cost (high level bands).

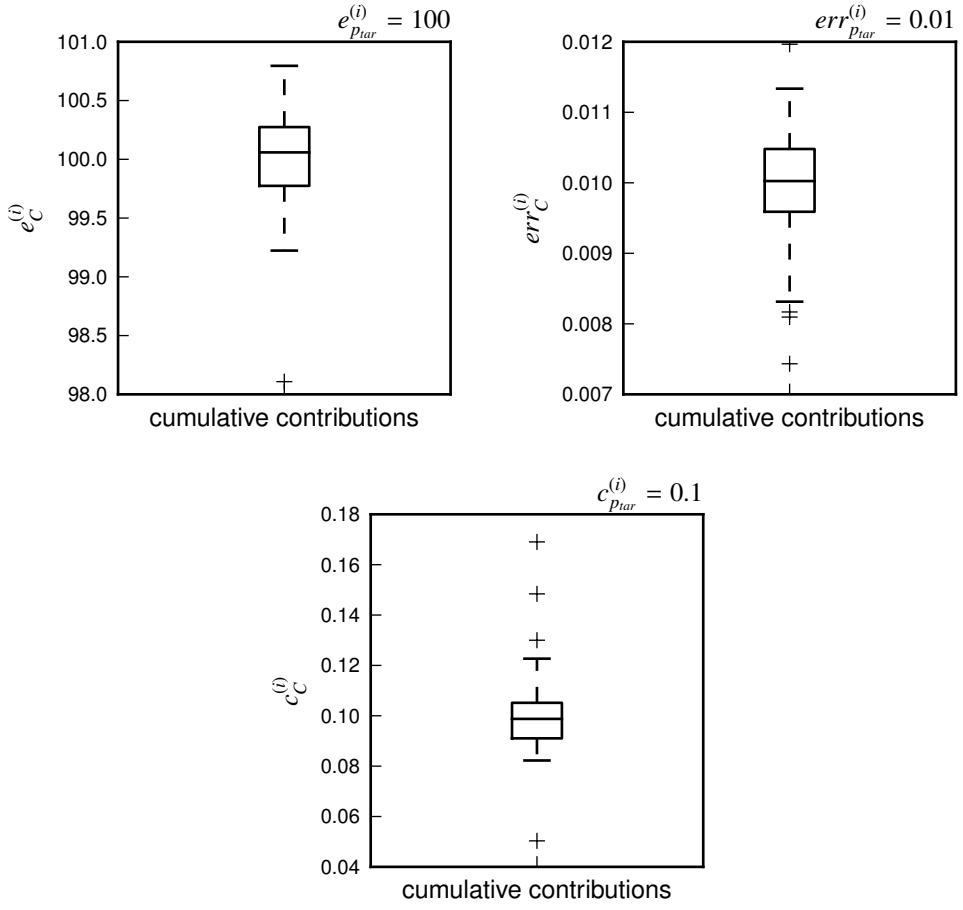


Figure A.2: Cumulative values of a single simulation run.

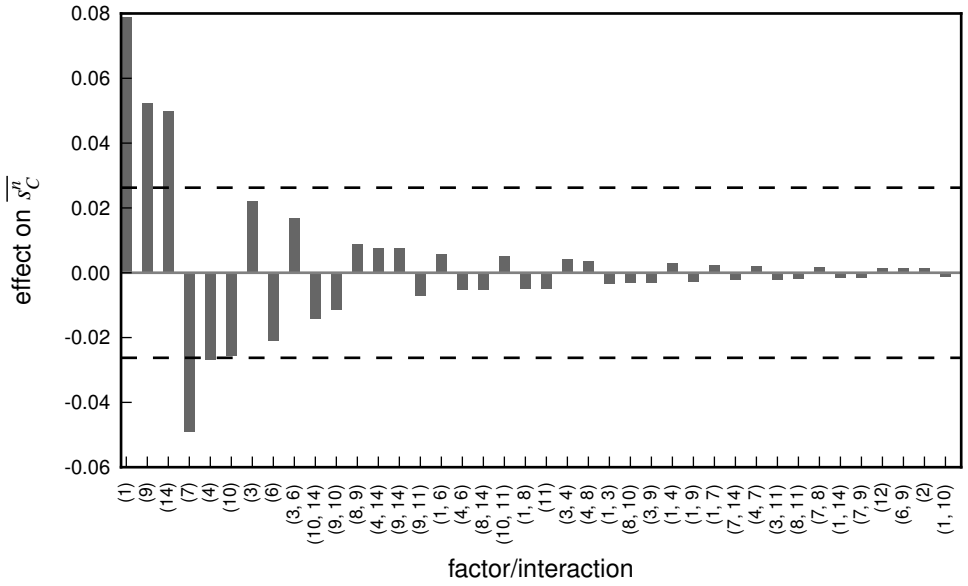


Figure A.3: Main and interaction effects on the mean normalized coalition scope $\overline{s_C^n}$.

B Supplements to Experiment *E-RQ6*

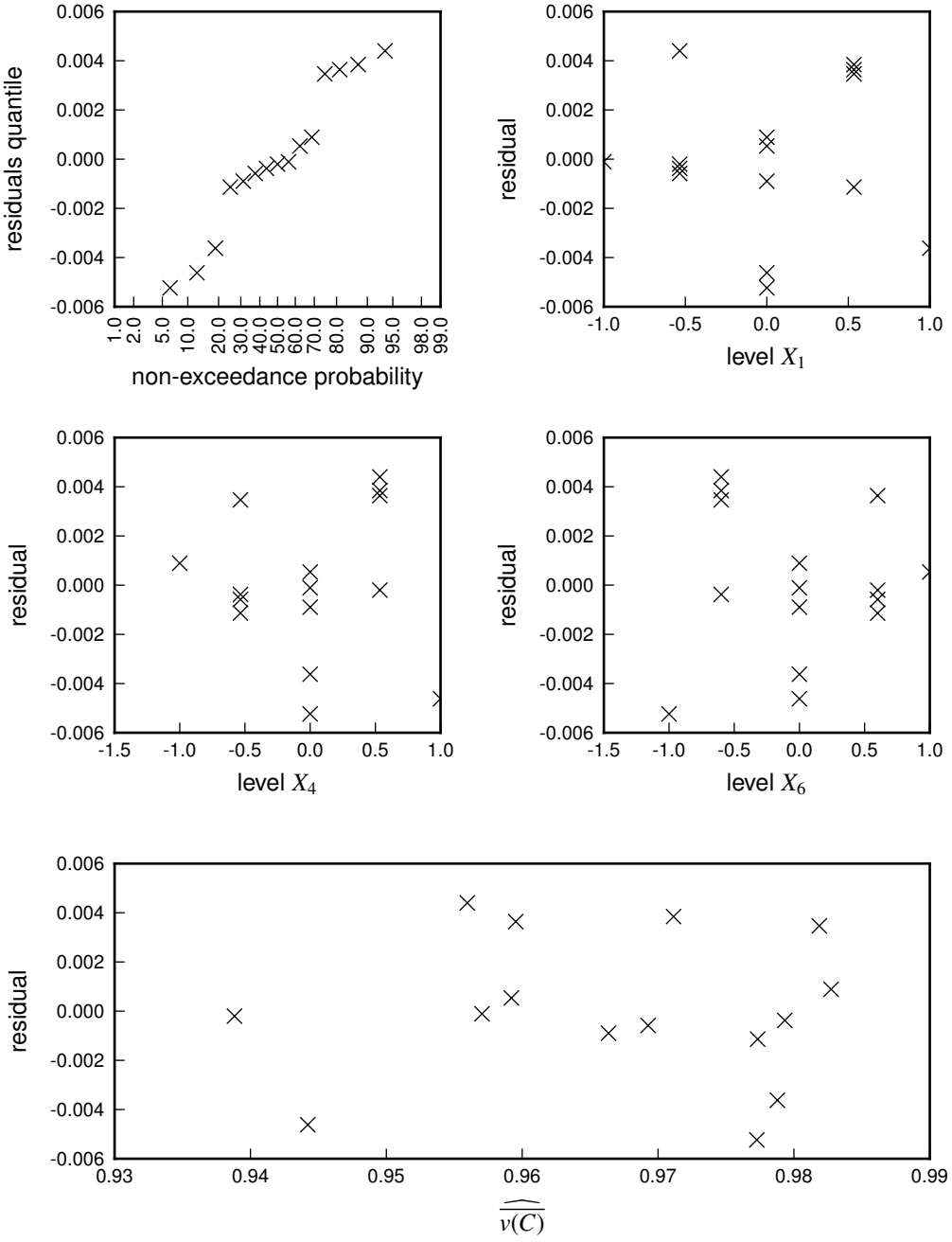


Figure B.1: Residual analysis of surface model $\widehat{\overline{v(C)}}$.

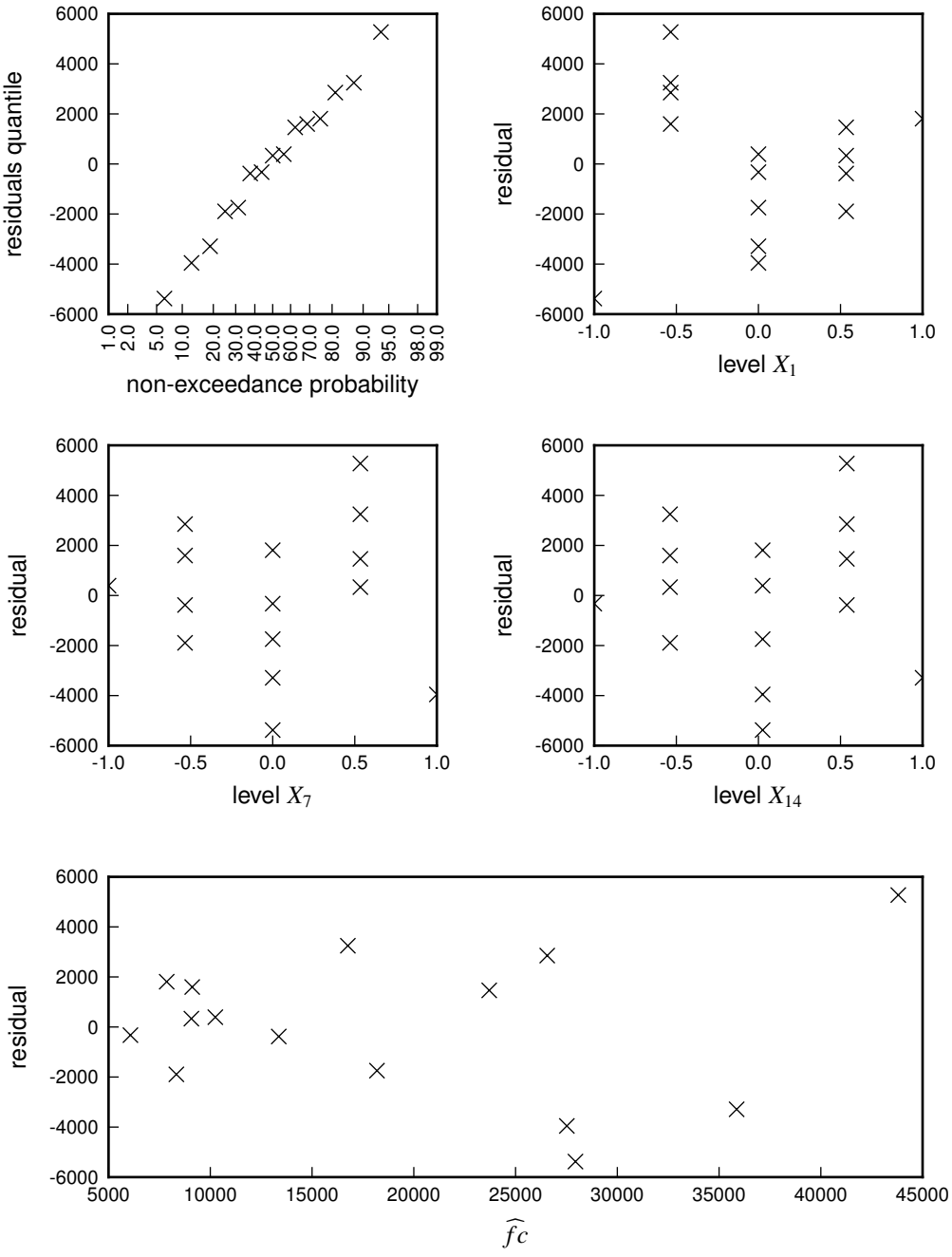


Figure B.2: Residual analysis of surface model $\widehat{fc}$.

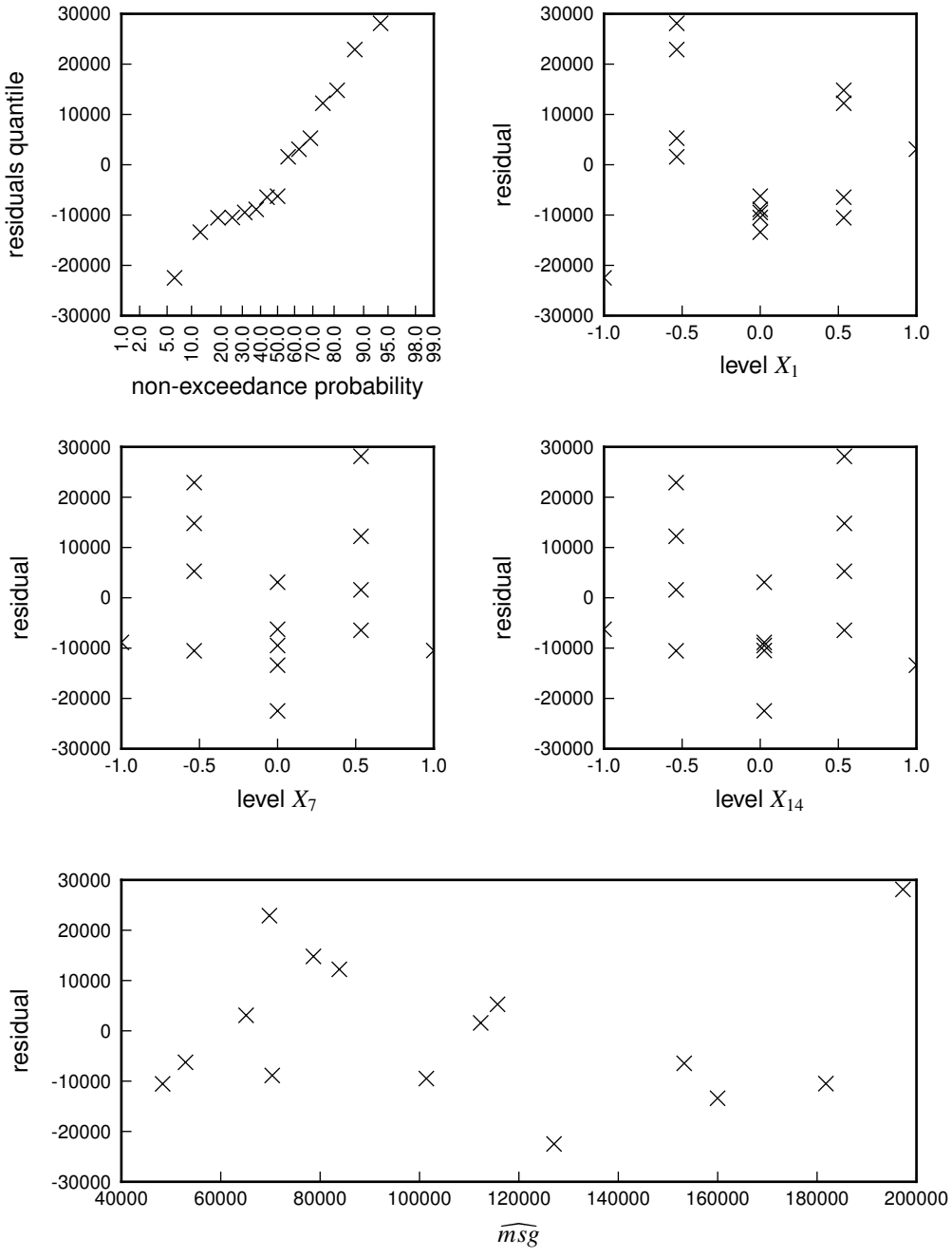


Figure B.3: Residual analysis of surface model $\widehat{msg}$.

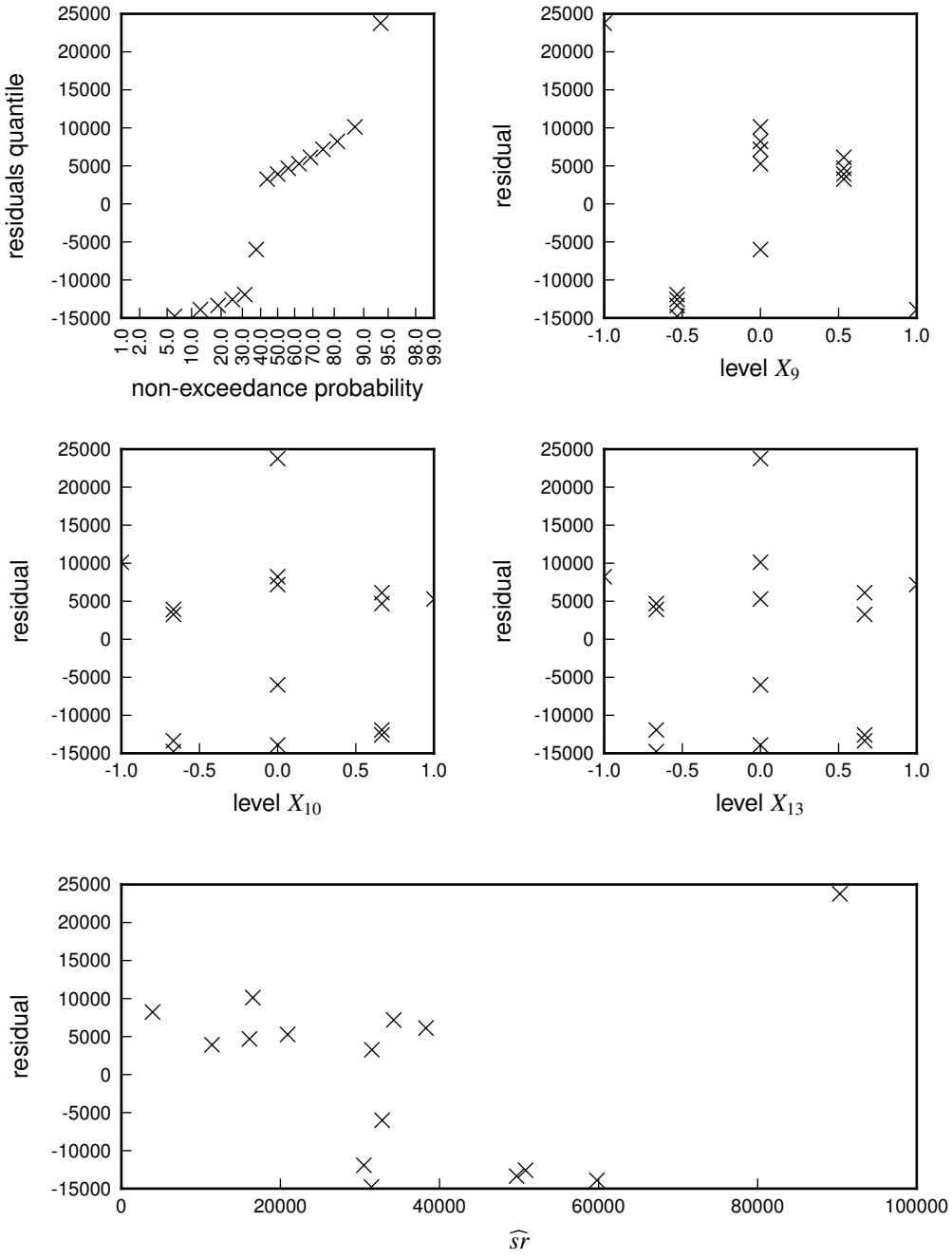


Figure B.4: Residual analysis of surface model $\widehat{sr}$.

ID	factor	level
X_2	$err^{(i)}_{p_{tar}}$	0.01
X_3	$c^{(i)}_{p_{tar}}$	0.1
X_5	$err^{(i)}_{p_{tar},-/+}$	0.005
X_7	$\|A\|$	1000
X_8	r_{init}	0.1
X_9	ext_N	10
X_{10}	$i_{u,max}$	2
X_{11}	$rCom$	10
X_{12}	t_{bi}	2
X_{13}	bi_{max}	8
X_{14}	oc_{max}	40

Table B.1: Settings of the fixed factors in experiment *E-RQ*6 (global performance).

ID	factor	level
X_2	$err^{(i)}_{p_{tar}}$	0.03
X_3	$c^{(i)}_{p_{tar}}$	0.1
X_4	$e^{(i)}_{p_{tar},-/+}$	8
X_5	$err^{(i)}_{p_{tar},-/+}$	0.01
X_6	$c^{(i)}_{p_{tar},-/+}$	0.1
X_8	r_{init}	0.1
X_9	ext_N	10
X_{10}	$i_{u,max}$	2
X_{11}	$rCom$	10
X_{12}	t_{bi}	8
X_{13}	bi_{max}	2

Table B.2: Settings of the fixed factors in experiment *E-RQ*6 (computational cost).

ID	factor	level
X_2	$err^{(i)}_{p_{tar}}$	0.01
X_3	$c^{(i)}_{p_{tar}}$	0.1
X_4	$e^{(i)}_{p_{tar},-/+}$	8
X_5	$err^{(i)}_{p_{tar},-/+}$	0.01
X_6	$c^{(i)}_{p_{tar},-/+}$	0.1
X_8	r_{init}	0.1
X_9	ext_N	10
X_{10}	$i_{u,max}$	2
X_{11}	$rCom$	10
X_{12}	t_{bi}	2
X_{13}	bi_{max}	2

Table B.3: Settings of the fixed factors in experiment *E-RQ*6 (communication cost).

ID	factor	level
X_1	$e^{(i)}_{p_{tar}}$	200
X_2	$err^{(i)}_{p_{tar}}$	0.01
X_3	$c^{(i)}_{p_{tar}}$	0.1
X_4	$e^{(i)}_{p_{tar},-/+}$	8
X_5	$err^{(i)}_{p_{tar},-/+}$	0.01
X_6	$c^{(i)}_{p_{tar},-/+}$	0.05
X_7	$\|A\|$	1000
X_8	r_{init}	0.4
X_{11}	$rCom$	10
X_{12}	t_{bi}	8
X_{14}	oc_{max}	1

Table B.4: Settings of the fixed factors in experiment *E-RQ*6 (runtime).

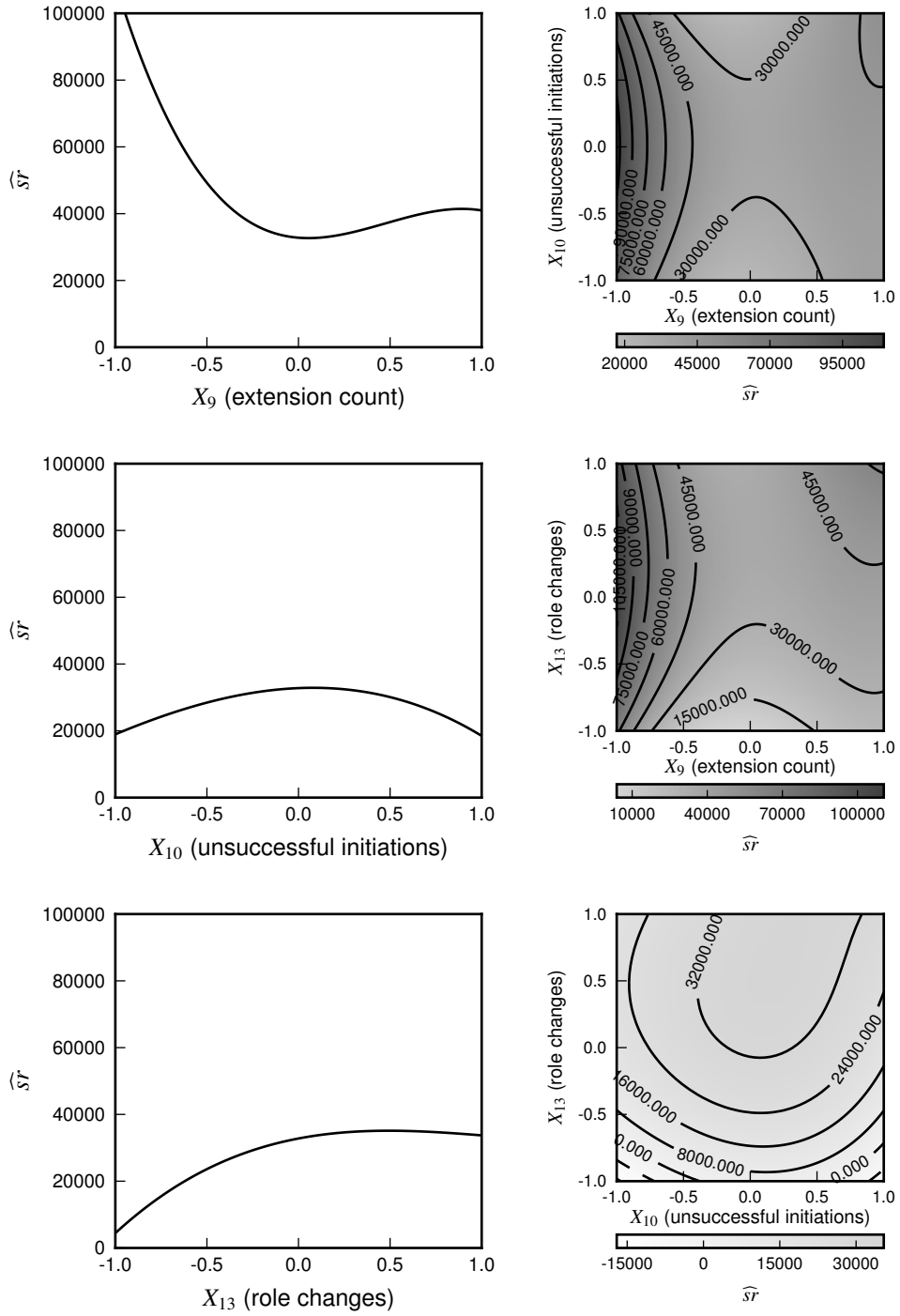


Figure B.5: Response surface of the simulated realtime (third-order polynomial).

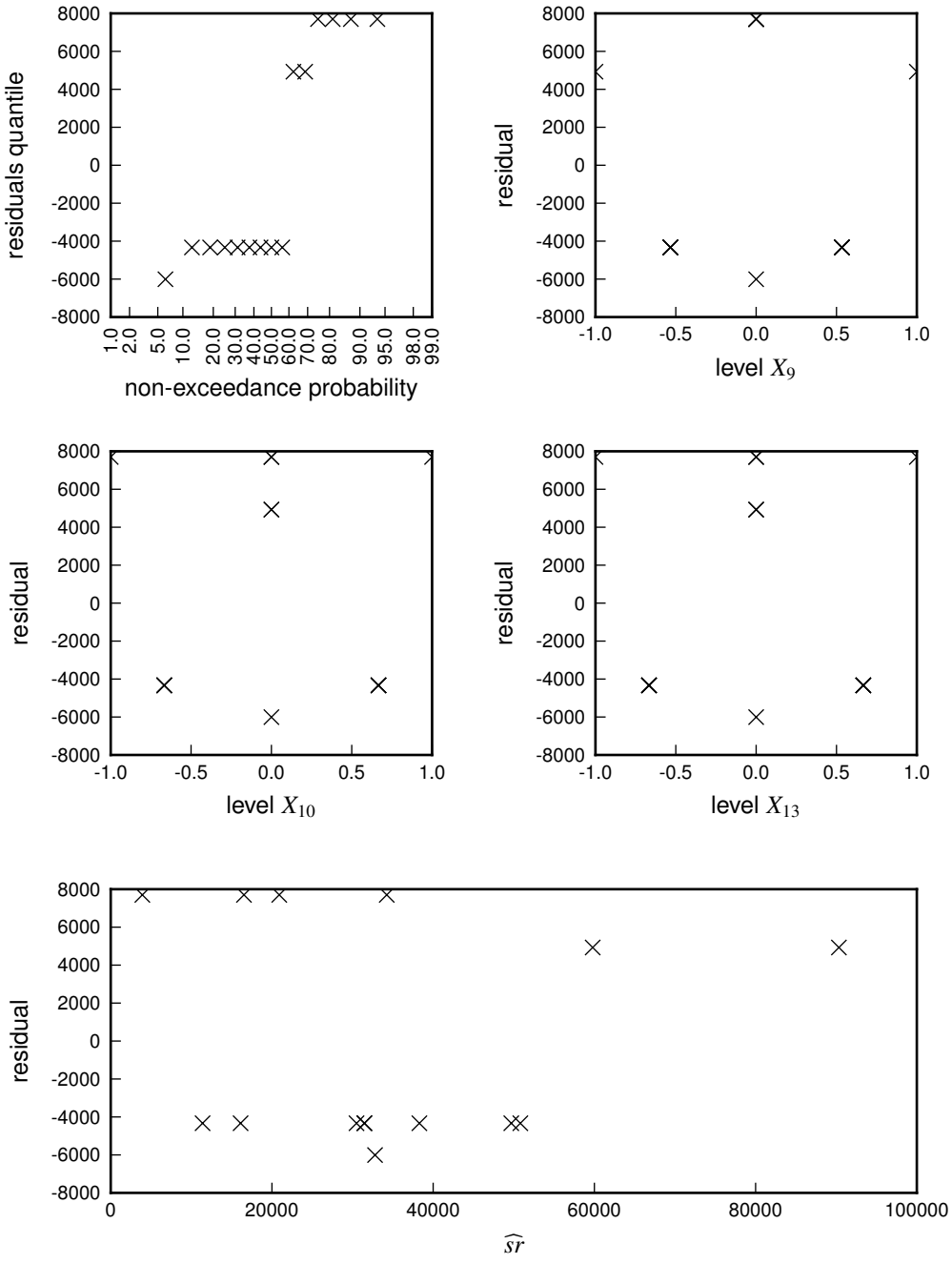


Figure B.6: Residual analysis of model $\widehat{sr}$ (third-order polynomial).

Glossary

In the following, the most important terms of this thesis are briefly explained, where a more detailed description is often given in the introducing sections. The explanations are generally based on own definitions as well as literature referenced in the text. With regard to the used symbols, ~ refers to the currently specified term, whereas ↑ references another term in the glossary.

Agent An ~ is an autonomous computer system which perceives its environment via sensors and acts upon it through effectors. It is considered intelligent if it is able to exhibit pro- and reactive behavior and interact with other entities. In the context of this thesis, an ~ controls a ↑unit and acts as its representative in an ↑electricity market.

Altruism ~ is a social attitude of an ↑agent. It is given if the agent chooses an action which does not maximize its individual benefit provided that it raises the benefit of another agent, the value of a ↑coalition, or the value of the global ↑coalition structure.

Central composite design A ~ is an ↑experimental design which is composed of a factorial design as well as star and center points. It is applied for the purpose of ↑response surface modeling.

Coalition A ~ is a set of ↑agents which cooperates in order to achieve a joint goal. In the context of this thesis, the latter is given by the fulfillment of a ↑product. A ~ is assessed by means of a ↑coalition value quantifying its worth. In the context of interactions with other parties, it is represented by a designated member which acts on behalf of the group.

Coalition structure A ~ is a set of ↑coalitions which partitions a set of ↑agents into disjunct subsets. The worth of a ~ is quantified through a ↑mean coalition value.

Coalition value A ~ is a measure for the worth of a ↑coalition. In the context of this thesis, it reflects the ability of a coalition to fulfill a ↑product.

Combined heat and power plant A ~ is a ↑power plant which converts a primary energy source like natural gas into ↑electrical energy based on a combustion engine, a fuel cell, or a gas turbine. The resulting heat is used for local supply in order to increase the energy conversion efficiency.

Contribution A ~ of an ↑agent to the ↑product of a ↑coalition is the amount of ↑electrical energy which is produced or consumed by its controlled ↑unit with a defined error at a defined cost within the ↑product horizon. Aggregating the contributions of all members of a coalition yields its ↑cumulative contribution.

Cumulative contribution The ~ of a ↑coalition is an aggregation of the ↑contributions of its members. Thus, it represents the cumulative amount of ↑electrical energy which

is produced or consumed by the members' ↑units with a defined cumulative error at a defined cumulative cost within the ↑product horizon of the coalition's ↑product.

Design of Experiments ~ is an approach for the performance of experiments based on ↑experimental designs which maximizes the gained information while minimizing the experimental efforts. Alternatively, ~ can be viewed as a domain providing concepts for efficient experimentation.

Distance ~ is a quantity which is used by an ↑agent in order to form ↑neighborhoods of nearest neighbors, where its calculation is based on a set of dedicated criteria.

Effect An ~ is the average change in a ↑response caused by the change of a single ↑factor or the combined change of multiple factors.

Egoism ~ is a social attitude of an ↑agent. It is given if the agent chooses an action which maximizes its individual benefit regardless of the resulting impacts on the other agents' benefits, the values of the given ↑coalitions, or the value of the global ↑coalition structure.

Electrical energy ~ is the flow of power over time and transmitted from a producer to a consumer by means of a dedicated ↑power grid. It is typically measured in kWh, MWh, or GWh.

Electricity market An ~ is a medium which allows actors to match supply and demand of ↑electrical energy through a bilateral or mediated trade of ↑products, where the place of delivery is given by a ↑power grid. With regard to their design, electricity markets differ in their degree of organization and centralization.

Energy ~ is the capacity to do work. It exists in different forms like ↑electrical ~ , thermal ~ , or kinetic ~ which are convertible into each other.

Experimental design An ~ is a list which lines specify different runs in a conducted experiment by determining the levels of the varied ↑factors.

Factor A ~ is a controllable or uncontrollable variable which influences a system. In the context of an experiment, controllable factors are systematically varied according to a given ↑experimental design in order to examine their ↑effects on a considered ↑response.

Factor screening A ~ is the identification of the most important ↑factors of a set of considered factors based on their ↑effects on a specific ↑response.

Full factorial design A ~ is an ↑experimental design which comprises all possible factor level combinations. It is applied for the purpose of a ↑factor screening.

ICT network An ~ consists of a set of communication nodes which are connected by means of communication lines. Each node is able to receive, process, and send data and can either be given by a separate computer system or an embedded system which is installed on a ↑unit.

Interaction An ~ is the mutual influence of multiple ↑factors on their ↑effects. The effect of an involved factor thus depends on the levels of the other factors, and vice versa.

Interaction effect An ~ is an ↑effect which is caused by an ↑interaction between multiple factors.

Main effect A ~ is an ↑effect which is caused by the change of a single factor.

Market role A ~ is a role which an ↑agent takes on in the context of a trade. It is determined by the ↑operation schedule of its ↑unit meaning that the agent either assumes the ~ of a producer or a consumer.

Mean coalition value A ~ is a measure for the worth of a ↑coalition structure which quantifies the average value of the comprised ↑coalitions.

Multiagent system A ~ is a system comprising multiple ↑agents. In order to achieve their goals, the latter communicate and coordinate their actions in a coherent fashion.

Neighborhood The ~ of an ↑agent is a set of agents lying within a maximum ↑distance from itself as quantified by a dedicated measure.

Operation hard constraint An ~ is a technical restriction of a ↑unit which has to be appropriately taken into account by an ↑operation schedule in order to be valid. ↑Operation hard constraint satisfaction can be verified through a corresponding satisfaction function.

Operation hard constraint satisfaction ~ is the fulfillment of an ↑operation hard constraint by an ↑operation schedule as indicated by a corresponding satisfaction function.

Operation schedule An ~ is a schedule which determines for each planning interval of a ↑planning horizon the amount of ↑electrical energy which is produced or consumed by a ↑unit with a defined error at a defined cost. It forms the basis for the ↑contribution of an ↑agent to a ↑product.

Operation schedule space An ~ of a ↑unit is the set of all ↑operation schedules satisfying its ↑operation hard constraints.

Operation soft constraint An ∼ is a target associated with the operation of a ↑unit which an ↑agent pursues to achieve through the selection of an appropriate ↑operation schedule in order to optimize utility. ↑Operation soft constraint satisfaction can be verified through a corresponding satisfaction function.

Operation soft constraint satisfaction ∼ is the fulfillment of an ↑operation soft constraint by an ↑operation schedule as indicated by a corresponding satisfaction function.

Operational flexibility ∼ is the property of a ↑unit to allow for a variable production or consumption within specific bounds. In general, the ↑operation schedule space of a flexible unit thus contains more than one feasible ↑operation schedule.

Payoff A ∼ is a monetary amount which a ↑coalition receives from or pays to another party for a traded ↑product. Its division among the members is determined by a ↑payoff distribution.

Payoff distribution A ∼ is a division of a ↑payoff which accounts for a set of distribution criteria. The individual shares represent the revenues or expenses which the members of a ↑coalition receive or pay after a successful trade of a ↑product.

Planning horizon A ∼ is a discretized time frame of consecutive planning intervals for which an ↑agent schedules the operation of its ↑unit in advance. Its length and temporal resolution may differ according to the given use case.

Power grid A ∼ is a network which consists of grid nodes being connected through power lines. Its purpose is the transmission and distribution of ↑electrical energy from producers to consumers.

Power plant A ∼ is a ↑unit which produces ↑electrical energy.

Product A ∼ is a specification which determines the amount of ↑electrical energy which is produced or consumed by a ↑coalition with a defined error at a defined cost in each product interval of the related ↑product horizon. It is termed a localized ∼ if its physical fulfillment refers to a specific section of a ↑power grid. A ∼ complies with a corresponding ↑product template.

Product hard constraint A ∼ is a restriction with regard to an attribute of a ↑product which has to be taken into account in the context of its specification. ↑Product hard constraint satisfaction can be verified through a corresponding satisfaction function.

Product hard constraint satisfaction ∼ is the fulfillment of a ↑product hard constraint by a ↑product as indicated by a corresponding satisfaction function.

Product horizon A ∼ is a set of possibly non-consecutive time intervals in which a ↑product is physically fulfilled.

Product portfolio A ~ is a set of target products which an ↑agent intends to fulfill within a considered ↑planning horizon through the formation of ↑coalitions. A target product is fulfilled if the values of a coalition's ↑cumulative contribution lie within the tolerance bands associated with its attributes.

Product soft constraint A ~ is a target associated with the specification of a ↑product which an ↑agent pursues to achieve in order to optimize utility. ↑Product soft constraint satisfaction can be verified through a corresponding satisfaction function.

Product soft constraint satisfaction ~ is the fulfillment of a ↑product soft constraint by a ↑product as indicated by a corresponding satisfaction function.

Product template A ~ is a set of ↑product hard constraints which a specification of a corresponding ↑product has to fulfill.

Rationality ~ is a property of an ↑agent. It is given if the agent always chooses an action which maximizes its expected utility taking into account all information available for decision making.

Reasoning scope A ~ is defined by the past experiences and future considerations which an ↑agent takes into account for decision making.

Regrouping A ~ is a repartitioning of two ↑coalitions in the course of coalition formation.

Regulatory hard constraint A ~ is a condition which has to be satisfied by an ↑agent when making decisions. ↑Regulatory hard constraint satisfaction can be verified through a corresponding satisfaction function.

Regulatory hard constraint satisfaction ~ is the fulfillment of a ↑regulatory hard constraint by an ↑agent as indicated by a corresponding satisfaction function.

Renewable energy ~ is an ↑energy which is extracted from a source being steadily replenished, like solar energy or wind energy.

Residual A ~ is the difference between an observed value and a value predicted by a model. Residuals thus allow to analyze the fit of a model.

Response A ~ is a measurable characteristic of a system which is possibly influenced by one or more ↑factors.

Response surface modeling ~ is the fitting of a polynomial function to experimental data, where the latter are typically obtained through the execution of a ↑central composite design or a Box-Behnken design. The resulting model describes a considered ↑response as a function of a set of ↑factors being represented by its variables. An analysis of the response surface thus allows for an identification of the optimal factor levels.

Solar power plant A ~ is a ↑power plant which converts solar energy into ↑electrical energy.

Strategy A ~ is a sequence of actions which is deliberately executed by an ↑agent in order to solve a task.

Trust The ~ of an ↑agent in another agent is a measure for the trustworthiness of the latter which has an impact on the willingness of the former to cooperate in the course of coalition formation.

Unit A ~ consists of one or more electrotechnical components which are connected to a node of a ↑power grid and able to produce or consume ↑electrical energy. It is connected to an ↑ICT network and controlled by an ↑agent which allows its potentials to be aggregated with those of other units through the formation of ↑coalitions.

Unit type A ~ is a category which reflects if a ↑unit is a producer or a consumer and provides ↑operational flexibility.

Value maximization ~ is the goal which ↑agents pursue when forming ↑coalitions. It can either concern the individual benefit of an agent, the value of a ↑coalition, or the value of the global ↑coalition structure.

Wind energy plant A ~ is a ↑power plant which converts kinetic energy into ↑electrical energy.

Bibliography

[Act00] "Act on Granting Priority to Renewable Energy Sources (Renewable Energy Sources Act)," 2000.

[ASSR12] G. Anders, F. Siefert, J.-P. Steghöfer, and W. Reif, "A Decentralized Multi-Agent Algorithm for the Set Partitioning Problem," in *PRIMA 2012: Principles and Practice of Multi-Agent Systems: 15th International Conference, Kuching, Sarawak, Malaysia, September 3-7, 2012. Proceedings*, 2012, pp. 107–121.

[AW07] M. J. Anderson and P. J. Whitcomb, *DOE Simplified: Practical Tools for Effective Experimentation*, 2nd ed. Productivity Press, 2007.

[BBRA10] S. Beer, L. Bischofs, H. Rüttinger, and H.-J. Appelrath, "Towards a Reference Architecture for Regional Electricity Markets," *it – Information Technology*, vol. 52, no. 2, pp. 58–64, 2010.

[BKM14] Z. Bodie, A. Kane, and A. J. Marcus, *Investments*, 10th ed. McGraw-Hill Education, 2014.

[BS14] J. Bremer and M. Sonnenschein, "Constraint-Handling with Support Vector Decoders," in *Agents and Artificial Intelligence: 5th International Conference, ICAART 2013, Barcelona, Spain, February 15-18, 2013. Revised Selected Papers*, 2014, pp. 228–244.

[BSD+09] R. Belhomme, M. Sebastian, A. Diop, M. Entem, F. Bouffard, G. Valtorta, A. De Simone, R. Cerero, C. Yuen, S. Karkkainen, and W. Fritz, "Deliverable 1.1: ADDRESS Technical and Commercial Conceptual Architectures," Tech. Rep., 2009.

[CCN+08] A. Costa, A. Crespo, J. Navarro, G. Lizcano, H. Madsen, and E. Feitosa, "A Review on the Young History of the Wind Power Short-Term Prediction," *Renewable and Sustainable Energy Reviews*, vol. 12, no. 6, pp. 1725–1744, 2008.

[CKY98] J. Contreras, M. Klusch, and J. Yen, "Multi-Agent Coalition Formation in Power Transmission Planning: A Bilateral Shapley Value Approach," in *Proceedings of the Fourth International Conference on Artificial Intelligence Planning Systems*, 1998, pp. 19–26.

[CLRS09] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. The MIT Press, 2009.

[CRK+11] G. Chalkiadakis, V. Robu, R. Kota, A. Rogers, and N. R. Jennings, "Cooperatives of Distributed Energy Resources for Efficient Virtual Power Plants,"

in *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems – Innovative Applications Track (AAMAS 2011)*, 2011, pp. 787–794.

[DGK05] G. Di Marzo Serugendo, M.-P. Gleizes, and A. Karageorgos, "Self-Organization in Multi-Agent Systems," *The Knowledge Engineering Review*, vol. 20, no. 2, pp. 165–189, 2005.

[Dir09] "Directive 2009/28/EC of the European Parliament and of the Council of 23 April 2009 on the Promotion of the Use of Energy from Renewable Sources and Amending and Subsequently Repealing Directives 2001/77/EC and 2003/30/EC (Text with EEA Relevance)," 2009.

[DP94] X. Deng and C. H. Papadimitriou, "On the Complexity of Cooperative Solution Concepts," *Mathematics of Operations Research*, vol. 19, no. 2, pp. 257–266, 1994.

[EGGW09] E. Elkind, L. A. Goldberg, P. W. Goldberg, and M. Wooldridge, "On the Computational Complexity of Weighted Voting Games," *Annals of Mathematics and Artificial Intelligence*, vol. 56, no. 2, pp. 109–131, 2009.

[FG97] S. Franklin and A. Graesser, "Is It an Agent, or Just a Program?: A Taxonomy for Autonomous Agents," in *Intelligent Agents III Agent Theories, Architectures, and Languages: ECAI'96 Workshop (ATAL) Budapest, Hungary, August 12–13, 1996 Proceedings*, 1997, pp. 21–35.

[Fou02a] Foundation for Intelligent Physical Agents, "FIPA ACL Message Structure Specification," 2002.

[Fou02b] ——, "FIPA Communicative Act Library Specification," 2002.

[Fou02c] ——, "FIPA Contract Net Interaction Protocol Specification," 2002.

[FWJ08] S. S. Fatima, M. Wooldridge, and N. R. Jennings, "A Linear Approximation Method for the Shapley Value," *Artificial Intelligence*, vol. 172, no. 14, pp. 1673–1699, 2008.

[HL04] B. Horling and V. Lesser, "A Survey of Multi-Agent Organizational Paradigms," *The Knowledge Engineering Review*, vol. 19, no. 4, pp. 281–316, 2004.

[HMPR04] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004.

[HS01] M. N. Huhns and L. M. Stephens, "Multiagent Systems and Societies of Agents," in *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, Ed. The MIT Press, 2001, pp. 79–120.

[HTWI09] L. Hu, G. Taylor, H.-B. Wan, and M. Irving, "A Review of Short-Term Electricity Price Forecasting Techniques in Deregulated Electricity Markets," in *2009 Proceedings of the 44th International Universities Power Engineering Conference (UPEC)*, 2009.

[Jen01] N. R. Jennings, "An Agent-Based Approach for Building Complex Software Systems," *Communications of the ACM*, vol. 44, no. 4, pp. 35–41, 2001.

[JIB07] A. Jøsang, R. Ismail, and C. Boyd, "A Survey of Trust and Reputation Systems for Online Service Provision," *Decision Support Systems*, vol. 43, no. 2, pp. 618–644, 2007.

[JP05] A. Jøsang and S. Pope, "Semantic Constraints for Trust Transitivity," in *Proceedings of the 2nd Asia-Pacific Conference on Conceptual Modelling - Volume 43*, 2005, pp. 59–68.

[JSW98] N. R. Jennings, K. Sycara, and M. Wooldridge, "A Roadmap of Agent Research and Development," *Autonomous Agents and Multi-Agent Systems*, vol. 1, no. 1, pp. 7–38, 1998.

[Ket95] S. P. Ketchpel, "Coalition Formation Among Autonomous Agents," in *From Reaction to Cognition: 5th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW '93 Neuchâtel, Switzerland, August 25–27, 1993 Selected Papers*, 1995, pp. 73–88.

[Kle08] J. P. C. Kleijnen, *Design and Analysis of Simulation Experiments*. Springer, 2008.

[Kle13] W. Kleppmann, *Versuchsplanung: Produkte und Prozesse optimieren*, 8th ed. Hanser, 2013.

[KSK10] J. K. Kok, M. J. J. Scheepers, and I. G. Kamphuis, "Intelligence in Electricity Networks for Embedding Renewables and Distributed Generation," in *Intelligent Infrastructures*, R. R. Negenborn, Z. Lukszo, and H. Hellendoorn, Eds. Springer, 2010, pp. 179–209.

[KSW13] R. R. Kories and H. Schmidt-Walter, *Taschenbuch der Elektrotechnik*, 10th ed. Europa-Lehrmittel, 2013.

[KWK+05] K. Kok, C. Warmer, R. Kamphuis, P. Mellstrand, and R. Gustavsson, "Distributed Control in the Electricity Infrastructure," in *2005 International Conference on Future Power Systems*, 2005.

[LFP99] Y. Labrou, T. Finin, and Y. Peng, "Agent Communication Languages: The Current Landscape," *IEEE Intelligent Systems and Their Applications*, vol. 14, no. 2, pp. 45–52, 1999.

[LSH+11] E. Lorenz, T. Scheidsteger, J. Hurka, D. Heinemann, and C. Kurz, "Regional PV Power Prediction for Improved Grid Integration," *Progress in Photovoltaics: Research and Applications*, vol. 19, no. 7, pp. 757–771, 2011.

[Lün12] O. Lünsdorf, "Selbstorganisation virtueller Geräte für das Lastmanagement von Kleinverbrauchern," Ph.D. dissertation, 2012.

[MMAC09] R. H. Myers, D. C. Montgomery, and C. M. Anderson-Cook, *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, 3rd ed. Wiley, 2009.

[MS95] S. T. March and G. F. Smith, "Design and Natural Science Research on Information Technology," *Decision Support Systems*, vol. 15, no. 4, pp. 251–266, 1995.

[Nii06] T. Niimura, "Forecasting Techniques for Deregulated Electricity Market Prices – Extended Survey," in *2006 IEEE PES Power Systems Conference and Exposition, 2006. PSCE '06.*, 2006, pp. 51–56.

[NIS13] "NIST/SEMATECH E-Handbook of Statistical Methods," 2013. [Online]. Available: http://www.itl.nist.gov/div898/handbook/

[NLJ96] H. S. Nwana, L. Lee, and N. R. Jennings, "Co-ordination in Software Agent Systems," *British Telecom Technology Journal*, vol. 14, no. 4, pp. 79–88, 1996.

[NRE10] "National Renewable Energy Action Plan in Accordance with Directive 2009/28/EC on the Promotion of the Use of Energy from Renewable Sources," 2010.

[NTS14] A. Nieße, M. Tröschel, and M. Sonnenschein, "Designing Dependable and Sustainable Smart Grids – How to Apply Algorithm Engineering to Distributed Control in Power Systems," *Environmental Modelling & Software*, vol. 56, pp. 37–51, 2014.

[OGZ08] A. Ockenfels, V. Grimm, and G. Zoettl, "Strommarktdesign: Preisbildungsmechanismus im Auktionsverfahren für Stromstundenkontrakte an der EEX," 2008.

[OR94] M. J. Osborne and A. Rubinstein, *A Course in Game Theory*. The MIT Press, 1994.

[Pow] "Power Agent Trading Competition." [Online]. Available: http://powertac.org

[PRS07] D. Pudjianto, C. Ramsay, and G. Strbac, "Virtual Power Plant and System Integration of Distributed Energy Resources," *IET Renewable Power Generation*, no. 1, pp. 10–16, 2007.

[RHJ04] S. D. Ramchurn, D. Huynh, and N. R. Jennings, "Trust in Multi-Agent Systems," *The Knowledge Engineering Review*, vol. 19, no. 1, pp. 1–25, 2004.

[RMWJ15] T. Rahwan, T. P. Michalak, M. Wooldridge, and N. R. Jennings, "Coalition Structure Generation: A Survey," *Artificial Intelligence*, vol. 229, pp. 139–174, 2015.

[RN10] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Prentice Hall, 2010.

[RSBC98] D. M. Rousseau, S. B. Sitkin, R. S. Burt, and C. Camerer, "Not So Different After All: A Cross-Discipline View of Trust," *Academy of Management Review*, vol. 23, no. 3, pp. 393–404, 1998.

[RUS14] C. Rosinger, M. Uslar, and J. Sauer, "Using Information Security as a Facet of Trustworthiness for Self-Organizing Agents in Energy Coalition Formation Processes," in *EnviroInfo 2014: 28th International Conference on Informatics for Environmental Protection*, 2014, pp. 373–380.

[SD81] R. G. Smith and R. Davis, "Frameworks for Cooperation in Distributed Problem Solving," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 11, no. 1, pp. 61–70, 1981.

[SGL03] M. Sims, C. V. Goldman, and V. Lesser, "Self-Organisation Through Bottom-Up Coalition Formation," in *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, 2003, pp. 867–874.

[Sha88] L. S. Shapley, "A Value for n-Person Games," in *The Shapley Value: Essays in Honor of Lloyd S. Shapley*, A. E. Roth, Ed. Cambridge University Press, 1988, pp. 31–40.

[SK98] O. Shehory and S. Kraus, "Methods for Task Allocation via Agent Coalition Formation," *Artificial Intelligence*, vol. 101, no. 1–2, pp. 165–200, 1998.

[SKL+10] J.-P. Steghöfer, R. Kiefhaber, K. Leichtenstern, Y. Bernard, L. Klejnowski, W. Reif, T. Ungerer, E. André, J. Hähner, and C. Müller-Schloer, "Trustworthy Organic Computing Systems: Challenges and Perspectives," in *Autonomic and Trusted Computing: 7th International Conference, ATC 2010, Xi'an, China, October 26-29, 2010. Proceedings*, 2010, pp. 62–76.

[SLA+99] T. Sandholm, K. Larson, M. Andersson, O. Shehory, and F. Tohmé, "Coalition Structure Generation with Worst Case Guarantees," *Artificial Intelligence*, vol. 111, no. 1–2, pp. 209–238, 1999.

[SLB09] Y. Shoham and K. Leyton-Brown, *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2009.

[SLBT15] M. Sonnenschein, O. Lünsdorf, J. Bremer, and M. Tröschel, "Decentralized Control of Units in Smart Grids for the Support of Renewable Energy Supply," *Environmental Impact Assessment Review*, vol. 52, pp. 40–52, 2015.

[Smi80] R. G. Smith, "The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver," *IEEE Transactions on Computers*, vol. C-29, no. 12, pp. 1104–1113, 1980.

[SPH12] W. Ströbele, W. Pfaffenberger, and M. Heuterkes, *Energiewirtschaft: Einführung in Theorie und Politik*, 3rd ed. Oldenbourg Verlag, 2012.

[Sta13] StatSoft Inc., "Electronic Statistics Handbook," 2013. [Online]. Available: http://www.statsoft.com/textbook/

[Sto02] S. Stoft, *Power System Economics: Designing Markets for Electricity*. Wiley-IEEE Press, 2002.

[Syc98] K. P. Sycara, "Multiagent Systems," *AI Magazine*, vol. 19, no. 2, pp. 79–92, 1998.

[Trö10] M. Tröschel, *Aktive Einsatzplanung in holonischen Virtuellen Kraftwerken*. OlWIR, 2010.

[TSCY01] M. Tsvetovat, K. Sycara, Y. Chen, and J. Ying, "Customer Coalitions in Electronic Markets," in *Agent-Mediated Electronic Commerce III: Current Issues in Agent-Based Electronic Commerce Systems*. Springer, 2001, pp. 121–138.

[vWS05] M. J. N. van Werven and M. J. J. Scheepers, "The Changing Role of Energy Suppliers and Distribution System Operators in the Deployment of Distributed Generation in Liberalised Electricity Markets," Energy research Centre of the Netherlands, Tech. Rep., 2005.

[Wei01] G. Weiss, "Prologue," in *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, Ed. The MIT Press, 2001, pp. 1–23.

[Wer14] R. Weron, "Electricity Price Forecasting: A Review of the State-of-the-Art with a Look into the Future," *International Journal of Forecasting*, vol. 30, no. 4, pp. 1030–1081, 2014.

[Wis16] C. Wissing, *Marktbasiertes Redispatch mit Flexibilitäten von Netznutzern für das Verteilnetz*. OlWIR, 2016.

[WJ95] M. Wooldridge and N. R. Jennings, "Intelligent Agents: Theory and Practice," *The Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152, 1995.

[Woo09] M. Wooldridge, *An Introduction to MultiAgent Systems*, 2nd ed. Wiley, 2009.

[YS01] J. Yamamoto and K. Sycara, "A Stable and Efficient Buyer Coalition Formation Scheme for E-Marketplaces," in *Proceedings of the Fifth International Conference on Autonomous Agents*, 2001, pp. 576–583.