Proactive Hybrid Learning and Optimisation in Self-adaptive Systems: The Swarm-fleet Infrastructure Scenario

Christian Krupitzer[a,*], Christian Gruhl[c], Bernhard Sick[c], Sven Tomforde[b]

[a] *Universität Hohenheim, Food Informatics & Computational Science Lab (CSL), Fruwirthstr. 21, 70599 Stuttgart, Germany*
[b] *Christian-Albrechts-Universität zu Kiel, Intelligent Systems, Hermann-Rodewald-Str. 3, 24118 Kiel, Germany*
[c] *Universität Kassel, Intelligent Embedded Systems, Wilhelmshöher Allee 73, 34121 Kassel, Germany*

Abstract

Context: Smart and adaptive Systems, such as self-adaptive and self-organising (SASO) systems, typically consist of a large set of highly autonomous and heterogeneous subsystems that are able to adapt their behaviour to the requirements of ever-changing, dynamic environments. Their successful operation is based on appropriate modelling of the internal and external conditions.

Objective: The control problem for establishing a near-to-optimal coordinated behaviour of systems with multiple, potentially conflicting objectives is either approached in a distributed (i.e., fully autonomous by the autonomous subsystems) or in a centralised way (i.e. one instance controlling the optimisation and planning process). In the distributed approach, selfish behaviour and being limited to local knowledge may lead to sub-optimal system behaviour, while the centralised approach ignores the autonomy and the coordination efforts of parts of the system.

Method: In this article, we present a concept for a hybrid (i.e., integrating a central optimisation with a distributed decision-making process) system management that combines local reinforcement learning and self-awareness mechanisms of fully autonomous subsystems with external system-wide planning and optimisation of adaptation freedom that steers the behaviour dynamically by issuing plans and guidelines augmented with incentivisation schemes.

Results: This work addresses the inherent uncertainty of the dynamic system behaviour, the local autonomous and context-aware learning of subsystems, and proactive control based on adaptiveness. We provide the 'swarm-fleet infrastructure'—a self-organised taxi service established by autonomous, privately-owned cars—as a testbed for structured comparison of systems.

Conclusion: The 'swarm-fleet infrastructure' supports the advantages of a proactive hybrid self-adaptive and self-organising system operation. Further, we provide a system model to combine the system-wide optimisation while ensuring local decision-making through reinforcement learning for individualised configurations.

Keywords: Self-Awareness, self-reflection, hybrid optimisation, autonomous learning, proactive behaviour, swarm fleet infrastructure, autonomous taxi, organic computing

1. Introduction

Imagine your autonomous car brings you to your job, picks you up at the end of your working day, and uses the time in-between to take care of maintenance or earns money. Given the current developments, autonomous driving seems to become a reality in a few years. Augmenting this behaviour with autonomous decisions about prioritising and fulfilment of tasks (i.e., serving as a taxi by transporting people or goods from one location to another), and maintaining the operational status themselves (i.e., control energy load and service status) is just the next step in this process. Although this may seem to be a utopian vision from our current point-of-view, the way to make it real is already paved, as shown by examples such as Elon Musk's vision for his company TESLA [1], Uber's activities[1] regarding self-driving robotaxis or the recent start of Waymo's robotaxi service in San Francisco[2]. Whereas current research and development approaches mainly target the technical feasibility of the driving itself, in such a setting, novel technological concepts are required that allow for improved efficiency of resource utilisation, decreased environmental impact of mobility, and improved user-oriented behaviour—while respecting the autonomy of the taxis and yet searching for a system-wide optimisation. This is in-line with the overall need to support the "sustainable development goals" postulated by the United Nations with concepts from the domain of artificial intelligence / machine learning [2].

We use this scenario to highlight a fundamental conflict in resource planning for smart and adaptive systems to which the class of self-adaptive and self-organising (SASO) systems[3] be-

*Corresponding author: phone: +49-711-459-23664; fax: -24433;
Email address: christian.krupitzer@uni-hohenheim.de (Christian Krupitzer)

[1] https://www.uber.com/de/en/atg/technology/

[2] https://techcrunch.com/2021/08/24/waymo-launches-robotaxi-service-in-san-francisco/

[3] We refer to SASO systems as an umbrella term combining the efforts in fields such as Organic Computing [3], Autonomic Computing [4], Self-Aware Computing [5], Interwoven Systems [6], or Autonomous Learning [7]

longs: Those systems support adjustments of their behaviour to changes in the environment [8]; however, a system-wide (or global) optimisation may result in the best possible trade-off between local goals, fairness, and self-awareness (as defined in [9, 5]) of environmental conditions but the autonomous participants may ignore the plan and consequently render the optimisation process infeasible. According to Kounev et al. a computing system is *self-aware* if it has or can acquire the following three characteristics: *self-reflective* (being aware of its architecture, execution environment, and operational goals), *self-predictive* (predict the effect of dynamic changes and adaptation actions), and *self-adaptive* (proactively adapting as the environment/system changes to met operational goals). Additionally, a global process is complicated by the ever-changing influences of the dynamic environment. On the other hand, a fully distributed scheme (i.e., no centralised control and only the autonomous taxis perform decisions) will neither identify globally optimal solutions nor close to optimal approximations due to communication and neighbourhood restrictions as well as the selfish behaviour of the subsystems [10]. This will further entail unfairness and decreased levels of service for people and goods at less prominent locations. Besides these aspects of the system organisation, the decision base cannot be handled traditionally: (1) the dynamics of constraints and goal functions introduce too frequent change, (2) reliable forecasts need to be considered to allow for proactive behaviour, and (3) we need to model and consider uncertainties to allow for appropriate decisions, where we need to distinguish between the inherent uncertainty of a problem and the uncertainty caused by insufficient information (in both aspects, aleatoric and epistemic).

This article extends the position paper from [11]. The major contribution is the vision of integrated system behaviour that combines the advantages of global optimisation with uncertainties and local self-improvement using autonomous learning techniques. We assume that such an approach results in several novel techniques and concepts that allow for establishing and operating the swarm-fleet-infrastructure (SFI—that infrastructure in which the autonomous taxis operate) as an example, while achieving three general goals: a) allowing for increased robustness against intentionally wrong or even faulty behaviour through flexible plans, b) an improved utility of both, the autonomous subsystems (i.e., the autonomous taxis) and the overall SASO system (here: the SFI) fulfilling especially goals of the local authorities as the possible operator, and c) a fast adaptation to changes in the characteristics of the environment and the learning problem (e.g., in terms of concept drift/shift).

Compared to the initial position paper, this article: i) provides an in-depth description of the swarm-fleet infrastructure introducing the different roles and responsibilities, ii) extends the use case towards a generic architecture for the design of hybrid SASO systems—combining central optimisation with distributed, intelligent decision making—including the definition of the particular components, and iii) presents an extended research roadmap with preliminary work that will serve as a basis to establish the vision of proactive, hybrid learning and optimisation system.

The remainder of this paper is organised as follows: Section 2 introduces the envisioned smart taxi fleet scenario and highlights that this is neither manageable by a centralised nor a distributed approach and requires proactive behaviour. Section 3 then presents our concept for a hybrid system optimisation that combines system-wide planning of possible behaviour corridors by the "swarm fleet infrastructure" (SFI) with autonomous learning of the most beneficial local behaviour and anticipatory state prediction explicitly modelling the inherent (un)certainties. To pave the path for this vision to become reality, Section 4 presents a research roadmap to establish such an integrated hybrid optimisation scheme in the sense of combining system-wide planning of behaviour with autonomous learning for distributed decision making. Section 5 discusses related work. Finally, Section 6 summarises the article and gives an outlook on future work.

2. Autonomous Taxis in the Swarm-fleet Infrastructure

In-line with the ongoing achievements in autonomous driving, the development of "autonomous taxi" (AT) services has been envisioned as a possible combination of different modes of transport, i.e., private cars, shared cars, and taxis [12]. The vision postulated by this article is that a user has on-demand access to a driverless taxi service but at a cost close to using a private car for the trip. This includes the assumption that future demands (i.e., as soon as a certain fraction of cars drives autonomously) for such trips are comparable to those of today or even higher.

We further assume that the traditional ownership and utilisation pattern will not be replaced completely by novel trends, i.e., the majority of cars is still owned by individuals having exclusive access to their car if needed [13]. This also includes the traditional working-hours and commuting model: A private-owned car is typically used for commuting to the job and back in the morning and afternoon peak hours on standard working days. In the remaining time, it is parked and not used. In addition, during bank holidays and weekends, the car is used for vacation trips and regular household business, such as shopping trips. While the working-day scenario follows an approximately static behaviour, the weekend and vacation profile is characterised by higher dynamics and the corresponding uncertainties. Consequently, this article initially focuses on the standard work-day profile.

Technically, we model the application scenario as an interplay of autonomous, distributed agents (the taxis) and an additional centralised service that does the accounting and guides the autonomous decisions of the agent. In particular, we argue that both traditional attempts to system modelling, i.e. fully centralised and fully distributed, are not appropriate due to several reasons. On the one hand, a centralised approach would assume that the SFI service is in charge of controlling the individual cars at decision-level (i.e., where to wait, which job to take, etc.). This would require constant communication of all internal state variables and perceived environmental conditions (which is not desirable due to privacy issues, primary goals given by the owner, and the locality of the decisions' impact, and it may have a negative impact on the acceptance of the system if the user loses control of its car), resulting in massive message load and high-frequency re-planning that is either not possible with current communication and computation infrastructure or not desirable due to the massive cost caused by such a scheme. Further, it reduces the scalability of the approach and introduces a threat to robustness since local behaviour has to wait for centralised decision to be taken until a response is possible – introducing delays that are not desired. On the other hand, a fully distributed approach without a centralised accounting and managing approach will lead to unfair distributions of jobs if agents behave uncooperative or egoistic. It will further neglect unpopular areas since jobs are too seldom to wait for them there or too far outside without explicit compensation. This may also have a negative impact on the service quality, the reaction times, and the reliability – with again implications for the acceptance (here at user-level). This means that – at least for the local authority – key requirements are hardly fulfilled, while individual taxi providers would benefit at the cost of the general public.

This section introduces the swarm-fleet infrastructure (SFI) as a centralised service environment in which the autonomous taxis fulfil their tasks (Section 2.1). This includes an introduction of all stakeholders of the overall SFI system. Afterwards, we focus on the privately-owned cars that serve as individual autonomous taxis (Section 2.2). Section 2.3 generalises the scenario in the context of hybrid SASO systems. Section 2.4 transfers the concepts to other application domains. Finally, Section 2.5 discusses threats to validity w.r.t. the scenario.

2.1. Centralised Perspective: The Swarm-fleet Infrastructure

The core of the SFI is an open service that is responsible for mapping customer demands (i.e., requests for taxi rides) to available resources. This should follow the goal of establishing fast and efficient provisioning of jobs, which also allows for customer satisfaction, fairness, and decreased congestion in traffic, for instance. We do not assume an owner model, i.e., the SFI can be operated by a commercial company (which results in a fee-based profit model), by local authorities (i.e., the city in which the fleet operates), or a non-profit foundation.

Our SFI is available as an open service. Customers can request transportation services that are published to all participating ATs. Furthermore, the SFI can apply sanctions and incentives to steer the behaviour (e.g., monetary incentives to wait in unattractive regions or decreased transport fees in very popular regions). The individual AT is then responsible for accepting/performing jobs, optimising its own operation (i.e., refuelling, maintenance, parking fees) and guaranteeing the service level of the owner. In more detail, the three basic tasks of the SFI are (see Figure 1):

1. Publish new jobs to the ATs: Provide the interface to the customers that can specify their rides and preferences, react with predicted service times, and manage the accounting. Please note that the SFI just enters the jobs and does not select the specific taxi—any taxi fulfilling the desired characteristics (e.g., number of passengers, comfort level, price level, transport capabilities), which is first entering the place is taking over the job.

2. Provide a continuously adapted plan for roles of taxis: Optimise waiting positions allowing for an improved coverage based on predicted demands, the current distribution of cars, and their capabilities.

3. Steering the autonomous behaviour using sanctions and incentives: Adapt the commission fees for types or locations of rides, pay compensation for waiting in less attractive regions, or manage tolls for roads.

Since the main purpose of the SFI is to establish an efficient and optimal mapping of customer demands to available resources, we can initially distinguish three stakeholders (as indicated by Figure 2): the customers, the cars, and the SFI.

Customers or clients are individuals or a group of individuals that want to book a drive. The customer participates in the SFI as a user of the service. The different clients have varying needs, e.g.:

- low waiting time (i.e., fast pick-up),
- short trip times,
- cost-efficient rides (i.e., low cost per trip),
- comfort needs (e.g., scenic route or smooth driving),
- transportation of goods (i.e., transportation space), or
- personal attributes (e.g., smoking permitted).

This is further combined with several possibly conflicting quality-of-service aspects including: i) preference for cars with high reliability, ii) preferences for luxury cars, iii) preferences for cars with high reputation or trust values, or iv) safe routes. Consequently, the customer should be able to define preferences and he needs to have access to real-time information about the service such as waiting time, travel time, and costs including possible alternatives to optimise the trip choice.

Taxis: We assume heterogeneous types of taxis within the SFI. This includes cars from different manufacturers, of varying sizes, or different engine types. We assume that each car can be modelled as a self-motivated agent that has a set of individual desires (e.g. high healthiness status, reputation, cost-efficiency, or reduced wear) and beliefs (i.e., a model of the environment, users, and the self based on observations). Such a car is owned by a private or commercial actor and is able to make its own decisions. We discuss the car as an autonomous taxi with its goals in the next subsection.

Swarm-Fleet Infrastructure: The SFI is provided and controlled by an operator. The goals are correspondingly reflecting the intentions of the operator (e.g., a local authority aiming at a fair and reliable taxi service or earning money through fees). Its main purposes are the acceptance, distribution, and billing of requests for trips. For this purpose, it offers an open environment in which anyone can participate (based on an initial registration). For each car, the SFI creates and maintains a profile with the required characteristics, such as a unique identifier, the current position, planned route, or the availability of the car. As a second purpose, the SFI is responsible for an optimised

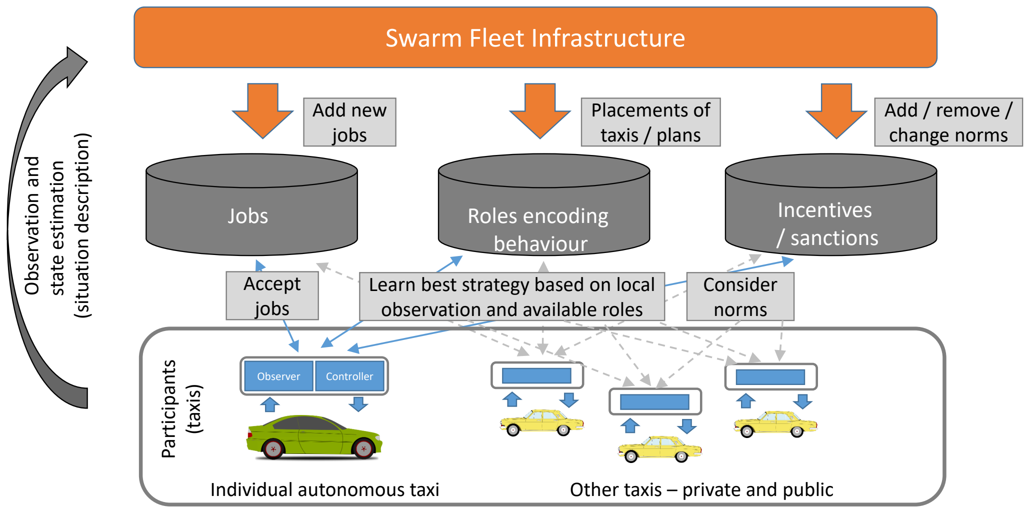

Figure 1: Interaction of players in the SFI. The overall SFI infrastructure is controlling the jobs and steers the system behaviour by providing optimised roles combined with incentives/sanctions issued via norms (in the sense of a norm-based agent system, [14]). The figure highlights an individual autonomous taxi (equipped with an observer/controller tandem as described in Section+2.2) acting in this infrastructure that has to compete and cooperate with other taxis.

and efficient planning of the taxi service. For this, continuous optimisation of waiting places as well as a role-based distribution of vehicles within the service region is carried out. This will be done proactively on the basis of forecasts of short and medium-term developments in order to achieve the goals (expressed as a utility function) as well as possible. As a direct control of the autonomous units is not preferable due to the interference with the autonomy of the individuals, the massive communication effort as well as the frequency of changes (both from the environment and by deviation of the autonomous units from the plan) is not possible, the SFI has intervention possibilities through incentives and sanctions [10]. These intervene in the settlement of the trips by paying compensation or withholding increased fees. This is made known in the system by means of norms and continuously adapted to changing circumstances.

Besides these three general stakeholders, the SFI approach comprises several other parties (according to [16]. We introduce the most prominent ones in the following.

Private owners: As mentioned above, we assume private-owned cars as the backbone of the SFI system. Consequently, we assume private individuals or households owning autonomous vehicles, and sharing them with the system during idle periods. The goal for these private owners of participating in the SFI comprises several (potentially conflicting) goals. The most important ones include earning money, provisioning of the car when needed, maximisation of the lifespan of the car, low administration effort (e.g., due to automated maintenance, repairs, parking, and charging), as well as keeping the car safe (avoidance of confrontations and malign customers).

Commercial taxi companies: Besides the main usage pattern of the SFI postulated by this article (sharing privately-owned autonomous cars as taxis), we further allow for commercial companies as operators of sub-fleets of autonomous cars. The cars within these sub-fleets should collaborate to maximise the owner's profit and avoid competitions within the peer group. Although these fleet operators may have more market power and a higher potential of optimising their profits (e.g., by more data and a better statistical foundation for planning and decision processes), we will not distinguish between owner models in the remainder of this article.

City/local authority: The AT participating in the SFI use the available urban traffic infrastructure. The local authority owning, operating, and maintaining this road network has the task to ensure the satisfaction and safety of their citizens. An array of subgoals can be inferred including minimal delays (i.e., no traffic jams), minimal pollution (i.e., exhaust, noise), minimised wear on infrastructure, guaranteed access for emergencies, avoidance of peaks in energy demand, and minimisation of re-structuring efforts. Consequently, the SFI-driven autonomous taxi service is to a certain degree a competitor of typically city-operated (or at least city-charged) public transport services. Consequently, an integration of both systems or coordinated behaviour is desired. However, such an AT service can also be used as an instrument for unburdening public transportation systems that are reaching the limits of their capacity as well as making their behaviour flexible.

Power supplies: Since we assume an extensive use of electrically powered vehicles, the charging cycles of the vehicles are elementary components of the system. Uncoordinated load peaks can occur at this point. Consequently, the local network operators have an inherent interest in balanced use of the energy networks through coordinated charging cycles (which is a current research topic as well).

Mechanics: Besides the taxi service, a second goal of the

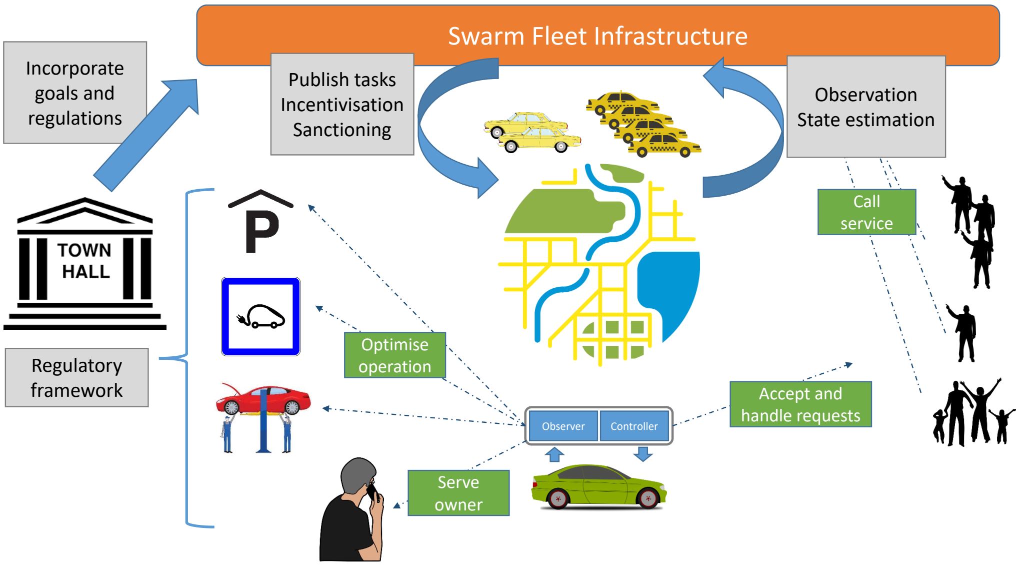

Figure 2: An autonomous car is equipped with an additional control module (here implemented as observer/controller tandem according to [15]) and operates within the SFI: It serves a user, optimises the operations (e.g., parking, maintenance, charging), and accepts and performs rides (in competition with others cars) offered by the SFI.

AT is to take care of maintenance and repairs if necessary. We summarise the stakeholders providing different garage options as 'mechanics'. Their primary goal is to earn money from offering these services, and this primary goal is supported by satisfied customers, minimised lineups, minimised emergencies, minimised drop-in times, great numbers of long-term contracts, and a maximised load. In addition, the business schedule needs to be aligned with the number of available employees and their working hours.

Other stakeholders: The previous list names the most prominent stakeholders and their goals. Of course, several further stakeholders with the focus on optimising traffic flows or commercial interest participate in the SFI system and impact its operation, such as event locations (e.g., stadium operators), car manufacturers, car dealers, parking space providers, or shop owners.

2.2. Decentralised Perspective: The Autonomous Taxi

Compared to the current usage pattern of privatly-owned cars, the benefit of participating in the SFI is the transformation of current "idle" phases (i.e., cars are not used for the majority of the day) into productive time. Technically, the autonomous and self-driving car is equipped with a self-motivated control unit realised as Observer/Controller tandem [15] following the terminology of OC [3]. However, alternative terminology may be used as well (e.g., 'MAPE-k' cycle in Autonomic Computing [4] or the LRA-M loop in Self-Aware Computing [5]). Independent of the wording, the concept is always the same: A management unit is added on top of the productive part (here: the car) to establish a robust and optimised self-adaptive and self-organising behaviour. In our case, the task of this autonomous unit (i.e., the Observer/Controller tandem of an individual autonomous taxi) is to earn money and to be available to the owner. In particular, the adaptation and planning logic of this component has to handle several (potentially conflicting) goals at the same time:

- At first, the availability of the car for its owner must be guaranteed, even if situations change. This reflects the traditional usage pattern, e.g., commuting to work and back, as well as spontaneous changes in these patterns.
- The first goal is to earn money as AT service brokered by the SFI. With the main purpose as constraint, this AT service is restricted to the local environment.
- The second goal is to take care of the required maintenance and re-fuelling autonomously. This includes scheduling charging cycles (i.e., assuming electric vehicles) with the corresponding cost profiles and availability of chargers (includes garage appointments).

Figure 2 illustrates an example of the integration of a specific car (in the lower middle of the figure)—which is assumed to be controlled by an on-board autonomic manager—into the SFI. Besides determining and accomplishing an optimised behaviour based on these goals, the car has to consider a set of secondary goals, for instance: i) the car should be clean (which renders the transportation of dirty goods unattractive or requires subsequent cleaning time), ii) it must maintain a given charging threshold to allow for non-stop rides, or iii) it should decrease parking and travel cost.

2.3. Hybrid SASO System Constellations

The SFI scenario is meant as a challenging scenario for a wider class within the domain of SASO systems that are characterised by specific characteristics. In this subsection, we initially specify the control problem in the SFI scenario to emphasise the conflict in goals between the system-wide operator and the participants. This further serves as a basis to generalise the concept towards a sub-class of SASO systems. To highlight such a general character, we discuss further application scenarios where the underlying characteristics of this sub-class need to be addressed as well.

Figure 3 presents a refined perspective on the control problems in the SFI scenario. Since this paper presents a first concept rather than a specific implementation, there is inherent uncertainty about the problem. In particular, the figure provides a first sketch of the control problem at the two different levels with the envisioned state, condition, and action variables as well the different components or services required that are to be analysed for defining the different components. Based on this, a more formal approach will be needed as soon as the scenario is realised.

The control problem of the centralised SFI takes observations from the environment and the traffic system that are combined into a situation description in the observer component (assuming a system architecture following the Observer/Controller tandem from the OC domain [15]). This information is then used to adapt the system behaviour at different time scales: Immediate task issuing and accounting is combined with long-term system behaviour optimisation using norms and roles. The figure names examples for the corresponding control parameters. On the other hand, the AT are fully autonomous in the sense that they are self-motivated and not subject to direct control of the SFI. Consequently, the figure lists the input variables describing the local situation of each AT as well as the internal status variables and the possible actions. Considering the two-level control problem, we face different characteristics that are typically not addressed in the standard SASO problems.

Openness: The system is by design open in the sense that participants can join in or leave at any time. This also includes that there might be attempts to veil or cover the individual identify or history.

Competition versus cooperation: There is competition between entities for maximising the individual benefit. However, cooperation would be beneficial w.r.t. global benefits (and maybe also for local ones). Although being uncooperative (i.e., not considering the abstract global plan provided by the SFI, agents can contribute to the overall benefit as they are taking on jobs. However, they have a negative impact on other aspects of the global goal such as a fair distribution). Still, entities have only a local view and are limited in perceiving a global state, which is required to possibly negotiate a fair assignment of tasks.

Limited information: This means that the perception is limited to local sensor information, possibly enriched with the information provided by the urban traffic system and local neighbourhood knowledge provided using, e.g., ad-hoc communication.

Dynamic environment: The control and decision problem is facing high dynamics of both, the environment as well as the optimal actions, which leads to high uncertainty and complexity. This requires a kind of "Interaction-awareness" in the terminology of Self-Aware Computing – the entities change the environment through their actions (pick up - customer/task does not exist anymore).

Proactivity: At both levels, i.e., the SFI operation and the autonomous taxis, the planning horizon is crucial for efficient and fast reactions, which requires proactive behaviour and reliable forecasts of the underlying developments. Proactive behaviour can help to decrease the reaction time as reactions options are calculated in advance for seamless adjustments without a delay.

Autonomy: The task-aware actions and the optimisation of behavioural strategies are fully decoupled and assigned to different entities: Local, taxi-based decentralised decisions and learning to self-improve the own behaviour and central planning combined with continuous optimisation. Although there is a need for centralised optimisation services based on a system-wide perspective, compliance with global rules cannot be guaranteed since each AT is fully autonomous. However, the overall system performance relies on mechanisms to enforce the desired behaviour. This means that partly the SFI urges the uncooperative participants to contribute to the goal achievement due to sanctions and incentives.

As a summary, we can state that the two standard ways to establish SASO systems are not possible in the SFI scenario:

- A fully decentralised system architecture is not possible, since the individual goals are not in-line with the system goal of an operator and a global view for optimising decisions is not available (due to vast communication efforts and limited communication speed as well as privacy considerations).
- A fully centralised system architecture is not possible, since individuals are by design autonomous and selfish, rendering centralised control infeasible. In addition, a fully centralised operation would be too complex, too slow to adapt to changes and come with a single point of failure.

2.4. Further Application Scenarios

The SFI scenario is meant as a challenging scenario for a wider class within the domain of SASO systems that are characterised by the common characteristics outlined above. To provide a basis to generalise the concept towards a sub-class of SASO systems, briefly discuss further application scenarios where the underlying characteristics of this sub-class need to be addressed as well. Since not all scenarios are directly mappable to the SFI scenario, i.e. some aspects are not fully visible, We distinguish between two perspectives again: non-competitive and competitive scenarios.

2.4.1. Non-competitive Scenarios

Logistics (one company): Here, the lorries are owned by one company and are typically not competing. However, assuming autonomous driving for lorries as well, such a scenario is covered by the SFI characteristics as follows: a) Instead of passengers, the autonomous lorries compete for transportation

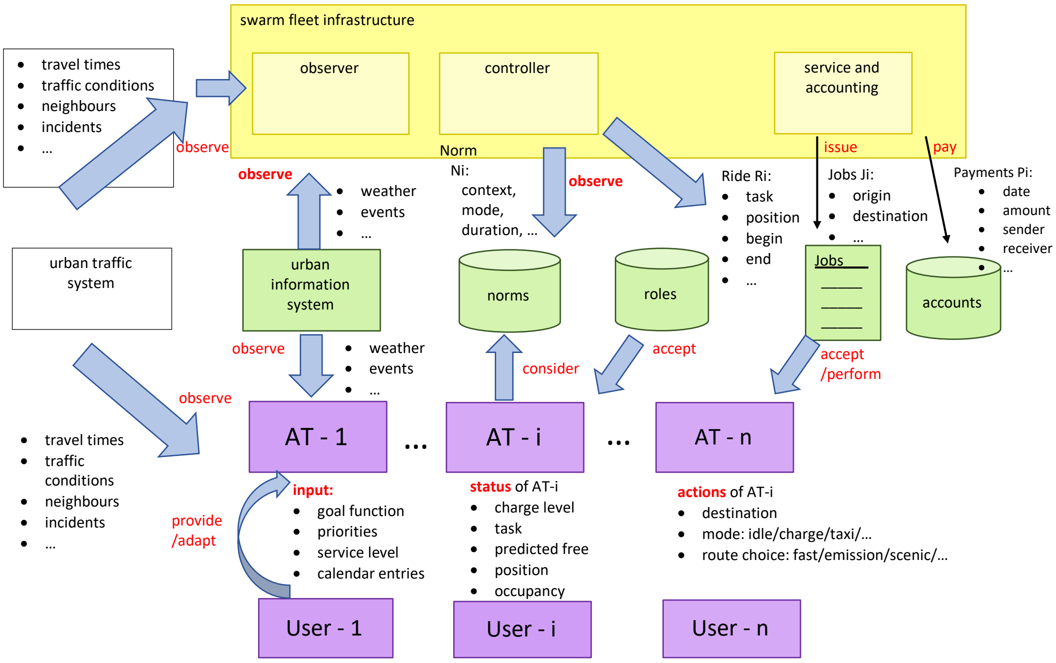

Figure 3: System model of the SFI system and the participating autonomous taxis. The figure shows examples for the different input and output variables as well as the status description of the taxis.

of goods (which can be turned into the same setting as the SFI), and b) the local behaviour of the lorries has to improve while taking guidance of the owning company into account that is done as a continuous optimisation and planning routine.

Platooning coordination: Platooning refers to the concept of coordinated driving to, e.g., safe fuel due to the slipstream effect. In such a typically motorway-based scenario, the self-motivated autonomous cars or trucks decide which platoon to establish or join while optimising their own benefit. In turn, a system-wide optimisation of platoons brings in the perspective to optimise the infrastructure efficiency, for instance.

2.4.2. Competitive Scenarios

Manufacturing: In Industrial Production, several plants of the same manufacturer or those of competitors can apply for the same processing steps of goods. In Industry 4.0 applications, this can be broken down into individual machines that have to autonomously organise themselves in groups and, consequently, also compete for resources.

Logistics: Already in current settings, large companies such as Amazon distribute delivery jobs to autonomous and independent drivers that compete based on characteristics such as reliability or locality. Although several aspects are similar to the SFI scenario, the major decisions and optimisation are done by the operator.

Pervasive Systems: Such systems assume the ubiquitous availability of devices. Here, competition is faced regarding variables of the shared environment, while the system setup combines autonomous units with a centralised system goal.

Smart grid environments: The introduction of autonomous, self-motivated units such as photo-voltaic and wind energy plants or electric vehicles and dynamic storage units are continuously transforming the previously centralised energy grid. In such a hybrid smart grid structure, we face the challenge of overall network stability (as a balance between power generation and utilisation as well as capacity limits), while the individual participants act self-motivated with limited knowledge.

2.5. Threats to Validity

The proposed scenario is presenting a possible use case for autonomous driving in the future, so details of the scenario are uncertain. One important aspect might be the existing traffic regulations. It might be possible that central coordination could be to some extent defined by law. Also, the autonomy of vehicles could be restricted, e.g., in the case of privately owned vehicles a limitation of the allowed travel distance per day or regulations about the number of people in a publicly available vehicle. On the other hand, it might be also possible the governmental actors can influence the local decision-making, e.g., by setting incentives for following the plan from the central instance.

This results also in uncertainties regarding the specific implementation; hence, we rather present the first concept in this paper. Accordingly, the system model is on a higher level of abstraction and will be finer conceptualised in the future. The same is true for the specific algorithms for prediction, determination of incentives, calculation of plans and activities and there like. Based on the next experiments, a more formal approach will be developed.

However, the mentioned example of Waymo in the introduction and activities of Uber shows that companies working on those topics and the transition into practice does not seem too far in the future. The future will show, how robust the assumption will be. Nevertheless, the scenario has merit for research in the mentioned constellation of SASO systems, which—as we have discussed above—can be found in many different application areas.

3. Hybrid Adaptation and Learning

The interplay between SFI and the set of ATs illustrates a very interesting challenge for future SASO system constellations that are characterised by

- openness (i.e., participants can join/leave at any time),
- competition (e.g., among the different ATs for jobs),
- dynamics (e.g., changing demands for taxi rides depending on the time of the day and depending on events such as a football match), and
- autonomy (i.e., a centralised SFI cannot tell the individual cars what they have to do—they decide on their own based on individual goals and utility functions).

Consequently, the SFI example illustrates that traditional approaches to system control as known in the SASO domain reach their end. Fully centralised planning and optimisation schemes will not work since the plan is obsolete as soon as the individuals do not follow the plan—although such a central planning unit is in general able to provide the best compromise (e.g., most efficient or fair) for all users. Fully decentralised decision schemes may converge to non-efficient and non-balanced (i.e., service times for customers in different regions of the city) solutions. Fully reactive solutions will not be able to react fast enough on the dynamics of the customer demands and the behaviour of the other participants in the system—and will probably fail to serve unusual events (e.g., customers leaving a football match or a concert) with an acceptable service level (measured, e.g., in waiting times).

To address these challenges, we generalise the example of the SFI as outlined above. We define a parallel process established by a system-wide Observer/Controller unit [15] as known from the Organic Computing domain [3] and distributed autonomous subsystems that are also equipped with their own Observer/Controller units. For the system-wide unit, we rely on the tasks and design as depicted by Figure 1. In addition to the illustration given in the figure, we assume that an observer component is responsible for gathering the environmental conditions including the states of the subsystems by fusing external sensor information (e.g., current location or traffic density) with reports of the subsystems. Determining the states of subsystems is augmented with forecasts of relevant system variables. Together with a quantification of the underlying uncertainty and detection of abnormal conditions, the generalised descriptions of the current conditions are reported to the controller unit that is responsible for self-improving its decision strategy, providing optimised plans for the system behaviour including possible behaviour of autonomous subsystems, and guiding the behaviour of the autonomous subsystems using norms (following the terminology of norm-based agent systems [14]).

Besides the SFI-based planning mechanisms, the autonomous units (i.e., the taxis) need an appropriate technical foundation to act successfully in the shared environment. Based on the Observer/Controller framework [15] as known from the Organic Computing domain [3], we define a multi-layered architecture for these autonomous subsystems. Figure 4 illustrates this architecture.

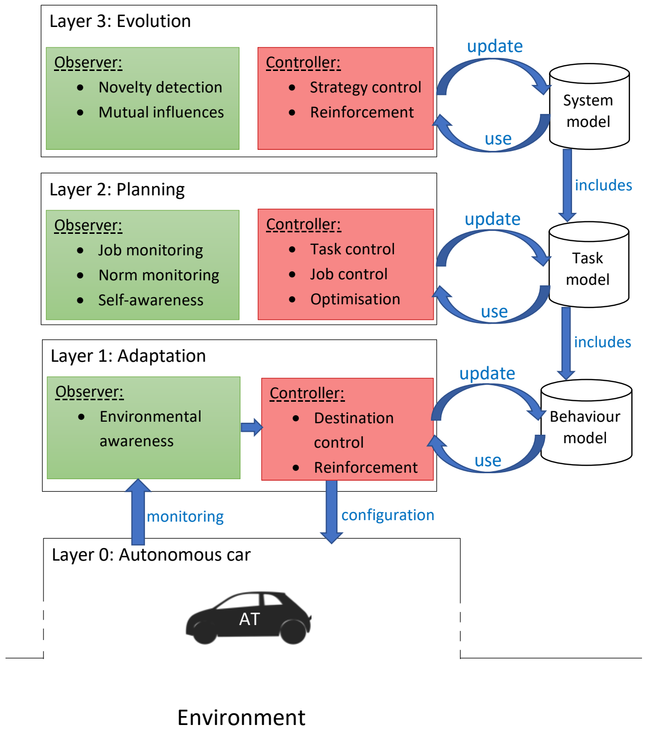

Figure 4: The architectural concept for realising the autonomous subsystems, illustrated using the example of an autonomous taxi. We distinguish between several layers that come with increasing time horizons. The usage of forecasts can allow for a better time-coupling of these layers. The resulting car with its three-layered architecture is embedded in the SFI infrastructure that – in general – adds an additional layer on-top. However, the upper layers of the taxi may also be performed externally, e.g. integrated into cooperative structures.

On the bottom layer (i.e., Layer 0), the productive part is integrated into the architectural concept. In the SFI scenario, this is the autonomous car that should serve as a taxi. This encapsulation requires the definition of interfaces for observing the internal and external conditions as well as modifying control parameters (e.g., current destination, route choice, etc.) according to [17].

Based on the encapsulation performed by Layer 0, Layer 1 is responsible for adapting the behaviour to current needs. Therefore, an observer unit establishes an environmental-awareness by providing a description of the current status augmented with forecasts of the expected behaviour. This serves as input to the controller that combines the decision control with reinforcement learning capabilities. The result of this process is a reconfiguration of the productive part using the control interfaces and an update of the knowledge base encode the most appropriate behaviour in the different conditions. We assume that evolutionary rule-based reinforcement learning techniques will play a key role to implement this behaviour, as e.g. outlined in [18] for urban traffic control or in [19] for smart cameras.

Since Layer 1 establishes an adaptation behaviour that acts synchronously with the productive part, Layer 2 augments this control loop with a planning component. Here, again an observer unit is responsible for monitoring the system-wide behaviour (e.g., expressed in the norms or by cooperation among cooperating subsystems). A self-awareness is established that assesses to which degree the current strategy is achieving the utility. In the SFI scenario, the observer also monitors the incoming jobs and filters them for promising candidates. In turn, the controller unit decides about the currently followed tasks (e.g., serving as a taxi or recharging in the SFI scenario) and the particular job (i.e., which customer to serve). This is combined with an optimisation that tries to find a trade-off between the conflicting goals. Based on that, a knowledge base is maintained that models the task behaviour.

Finally, Layer 3 provides means for long-term system evolution. A detection of novel process and abnormal conditions serves as an indicator if anything changes fundamentally (e.g., novel customer demands, new competitors, changed street topology). These indicators serve as input to the controller unit that maintains the overall strategy (encoded in a system model) followed by the subsystem, which is also subject to reinforcements. Here, also concepts such as (collective) self-reflection are established that identify inappropriate or missing knowledge, e.g., according to [16].

Several concepts exist that might be integrated into the discussed use case, for example the architecture (based on [15]), a concept for the SFI use case [11], and basic SASO mechanisms, e.g., for coordination of entities [10, 20], achieving fairness [21], or optimisation-based adaptation planning [22]. However, the described hybrid learning and optimisation system requires research efforts for the conceptualisation of novel techniques and approaches. In the following section, we outline the most urgent challenges in this context.

4. Primary Research Challenges

The concept presented above demonstrates that the vision of proactive (through the integration of forecasts) hybrid learning and optimisation requires combined efforts from different research directions. We need a consolidated and integrated approach that combines the current and upcoming research insights from three different core fields: a) self- and environment awareness in SASO systems, b) distributed autonomous learning of individual entities in shared environments, and c) centralised planning and optimisation. In addition, several further fields will have to provide valuable insights and techniques as soon as the core approach has shown its advantages in appropriate settings, these secondary fields include i) security, ii) communication, iii) computational trust, iv) incentivisation, and v) user-centric behaviour to name just the most urgent ones. Furthermore, these research efforts have to be accompanied by research on a near-to-reality testbed to demonstrate and analyse the behaviour in detail. Following the discussion above, we propose to use the scenario of AT services as a basis for such a testbed. In this section, we present a research roadmap towards investigating and implementing such an approach that distinguishes the efforts according to the above-introduced fields.

4.1. Self- and Environment-awareness in SASO Systems

As a first building block, we need to develop and investigate models that describe the local (i.e., AT-based) and global (i.e., SFI-based) state. We propose to make use of probabilistic models that are based on parameter estimation techniques reflecting the uncertainty arising from the observations. In this context, we have to identify model inputs and outputs relevant for the individual cars and the SFI (i.e., according to Figure 3). We need online parameter estimation techniques (i.e., online-learning) that consider the timeliness of observations and knowledge. We may assume an initial, prototypical model being available that has to be "customised" to one or a set of operation states regarded as being normal. Based on this expected behaviour model of the system, we focus on key techniques that make an AT self-aware. In contrast to [9] we further distinguish between dynamics/changes of the system itself (self-awareness) and its operational environment (environment-awareness) – primarily to have two simpler models (one for the system, one for the environment) instead of one complex model. Deviations from a normal state (e.g., other service behaviour due to dynamic customer constellations, changed owner demands, increase of fuel cost, or a fault of the AT) must be detected to trigger a self-adaptation using the aforementioned methods [23]. To detect changes—called concepts shifts or drift in the field of stream learning [24]—we rely inter alia on techniques from the field of Gaussian processes. The trade-off between timeliness and correctness of reactions has to be investigated in detail as temporal performance is frequently ignored in current research. Adaptation steps can also be triggered by the observer (internally) or the SFI (externally), e.g., due to changed goal functions [25].

A second building block covers research on methods for forecasts of the state of the individual taxi and the overall SFI (including customer demands). The focus is on probabilistic forecasts [26] with well-calibrated neural network models reflecting the confidence in estimates based on uncertainty in inputs and available data [27, 28, 29]. These uncertainty values then serve as a basis to assess the reliability of the forecasts—which is used to decide to which degree the predicted developments are taken into consideration or the current conditions are used.

4.2. Distributed Autonomous Learning

A core element of the autonomous taxis is the ability to act locally and to improve this behaviour over time. Due to the vast situation and action spaces for the individual taxis, we focus on reinforcement learning techniques that are specially designed to generalise situations without the need for enormous training data: learning classifier systems. Especially variants of Wilson's Extended Classifier System (XCS) [30] are promising candidates due to their generalisation capability and the differentiation between accuracy and strength. The local learning approach (i.e., at each autonomous vehicle) requires a variant of

XCS capable of learning the most efficient strategy—which includes dynamic search spaces, controlled exploration, and real-world restrictions (e.g., safety considerations). In contrast to standard XCS systems, we need to incorporate multiple goals and switch between them online, which requires novel encoding and credit assignment schemes of the gathered knowledge. Due to the limit availability of feedback, we need to introduce mechanisms for efficient knowledge exploitation (e.g., interpolation, transfer, or collaborative approaches). In turn, powerful techniques based on Deep Neural Network technology can be used to tackle the learning problem. However, they lack interpretability of their knowledge and explainability of their behaviour rendering a direct utilisation in scenarios where taxis act on behalf of humans as less desirable. However, a possible direction of research is to combine the advantages of XCS with deep learning concepts, as proposed in [31].

Based on the self- and environment-awareness mentioned above, the individual autonomous taxi can reflect about its own knowledge, as e.g. outlined in [16]. Especially when being embedded in a larger constellation (e.g. a community of trusted individuals [32]), a collective perspective can identify outdated, missing or even wrong knowledge. This can be combined with an opportunistic consideration of dynamically available knowledge sources (e.g., the owner, users, other cars, or unstructured data from the Internet) [33]. Both directions require a possibility to assess the characteristics of their own knowledge, which is perfectly in-line with the concepts of smart and adaptive systems.

In general, the task of learning the locally optimal behaviour is closely related to the field of multi-agent reinforcement learning (MARL) [34, 35]. For an introduction to the field, a brief clarification of terms and a detailed taxonomy is given by Busoniu et al. [34], for instance. Considering this taxonomy, the learning problem defined in this article is related to fully competitive and mixed games in both variants, static or dynamic. Since one possible way to model the global payoff is by defining a function of the local agents' payoffs, it can also be considered as a fully cooperative game for those taxis that belong to the same authority or are pooled in some way. In particular, the idea of modelling the learning problem lies in the approach to turn it into a Markov Decision Problem and to represent the various possible states of the individual taxis in such a process. This can be then solved by different kinds of reinforcement learners that are either backed by technology for longer-term predictions or the different methods summarised under the term self-awareness above.

Also in [34], the authors state that there is a severe complexity resulting from coordination in MARL scenarios – which is an open issue and a major issue in MARL systems. Therefore, novel learning concepts focusing on an online adaptation of input and action spaces as well as a dynamic complexity reduction of the learning problem are required - which we claim as being one of the major research challenges in the context of autonomous learning. For instance, a selection of those criteria and entities that need to be considered in the learning problem at runtime may be beneficial, e.g. following the methodology proposed by [36].

4.3. Centralised Planning and Optimisation

For the optimised SFI-based planning, we require fast, anytime algorithms that allow for determining corridors of behaviour rather than pre-defined solutions. The complexity of the large-scale optimisation problem renders exact solutions infeasible—consequently, it has to be investigated which heuristics are particularly applicable to identify 'good enough' solutions. Additionally, the optimisation problem then has to be extended to consider constraints (multi-objective optimisation including, e.g., a load of roads, pollution state, priorities given by local authorities) and overall system control figures (e.g., the budget of incentivisation). Further, we strive for many-objectiveness, i.e., fulfilling various individual goals simultaneously.

For tackling those challenges, we can build on previous works. In [22], we showed that the choice of a heuristic additionally requires to take the current situation and constraints (e.g., the convergence of algorithms) into account to build a situation-aware choice of the heuristic for optimisation. In the SFI scenario, this is additionally complicated as the environment is highly dynamic and unstable. For taming this uncertainty, the optimisation requires degrees of freedom. Using a learning approach helps to reduce the ranges for those degrees of freedom. Additionally, the planning has to balance the costs and investments as well as the earnings to provide a fair solution. In [21], we compare several compensation approaches to balance investments in coordination. Those approaches rely on trying to equally share negative impacts. However, in the SFI scenario, the planning requires to equally share the earnings. Both dimensions must be represented by the optimisation function. Additionally, a cooperation perspective [10] of the taxis or approaches that focus on interacting systems to compete for shared resources [37] can be studied.

In the ideal case, entities might behave altruistically [10], even in case the global optimal adaptation plan decreases their individual utility. As the autonomous resources follow their own—potentially conflicting—objectives, the integration is additionally challenging as those behave self-ish and competitively. Consequently, resources that are disadvantaged by the cooperation might potentially reject participation. To tackle these issues, we described in [10] different coordination mechanisms that can help to identify a solution for cooperation that balances the disadvantages across several instances and increase the motivation for cooperation: Those coordination mechanisms can be categorised into (i) selfish behaviour, (ii) altruistic behaviour, (iii) negotiation, (iv) enforcement of central decision making, and (v) rewards/incentives. Especially relevant in the context of the SFI scenario are rewards and incentives, but also enforcement of central decision making through sanctions. We plan to study the issues from such situations, including the identification of (i) mechanisms to reward entities for decreased utility and (ii) interaction mechanisms to control the behaviour of the autonomous taxis through sanctions.

4.4. Proactive Adaptation

The time of adaptation is a central question [8]. From the user's point of view, proactive adaptation is preferable, since it avoids interruptions in the user's workflow with the system. On the other hand, the prediction algorithms needed for proactive adaptation have several issues. They are complex to develop, their suitability is highly dependent on the specific prediction tasks, and faulty results can cause suboptimal or malicious adaptations. Therefore, many approaches focus on reactive adaptation [8]. However, the choice for proactive or reactive adaptation is not exclusive. Mapping the adaptation process to the Autonomic Computing MAPE cycle [38], the basic functionality for adaptation is monitoring the environment, analysing for change, computing adaptation plans, and executing these plans. Reactive and proactive adaptations involve similar activities regarding monitoring, planning, and executing, but strongly differ in the analysing phase. With reactive adaptation, the monitored data is analysed for abnormal patterns. With proactive adaptation, the monitored data is used to forecast system behaviour or environmental state. From the user's point of view, this is preferable, as it reduces interruptions and adaptations can be optimised for a sequence of events [37]. Further, proactive adaptation includes context adaptation via actuators in order to avoid unwanted situations. It is possible to combine proactive and reactive adaptation such that proactive adaptation is the goal, and reactive adaptation is used as a backup mechanism, i.e., if a change was not predicted (e.g., failing of a component).

In the SFI, we follow a hybrid approach w.r.t. the time dimension. The central element acts proactively, as the calculated plans integrate predictions of the future system and environment conditions as well as assumptions about behaviour. The decentral elements react to the situations, using the knowledge from the plans of the central entity. However, proactive adaptation has several challenges, especially in scenarios in which multiple systems share the context/environment. It is highly dependent on the correctness of the predictions, as faulty predictions can cause suboptimal adaptations. The major challenges here are predicting the time of an event with high enough accuracy, as well as predicting user behaviour and rare events, especially taking the incompleteness of the information into account. In the SFI scenario, especially the behaviour of the local entities are an important factor that needs to be taken into account.

4.5. Testbed

As outlined in Section 5.2, related work focuses on simulations of autonomous taxis by mostly replacing traditional taxis operated as fleets of a few large-scale vendors with autonomous taxis operated with the same owner model. These simulations are typically very abstract models of city environments using MATSim[4] [39], for instance. In order to be able to investigate the effects of traffic conditions as well as novel utilisation patterns (e.g., already caused by the owner's commuting pattern and the subsequently limited availability schedule), we propose to focus on realistic traffic topologies and demands.

[4] `https://www.matsim.org/`

We propose to use the 'Organic Traffic Control' (OTC) system [18] as a basis for developing an SFI simulation since OTC already provides a realistic traffic simulation controlling (which is missing in the state-of-the-art for AT services, see Section 5). OTC adapts and improves the efficiency of the green duration of traffic lights locally at each intersection [18], coordinates the intersection controllers without centralised elements to form progressive signal systems [40], and routes streams of vehicles based on a per-hop basis [41]. The distributed controllers communicate with each other via messages, e.g., using standard Internet protocols such as TCP/IP. The system is based on an integration of the professional traffic simulator 'Aimsun Live' [42] that is widely used by traffic engineers world-wide. It relies on real-world topology models (see Figure 5 for an example intersection located in Hamburg, Germany) and the corresponding actual traffic demands and control strategies.

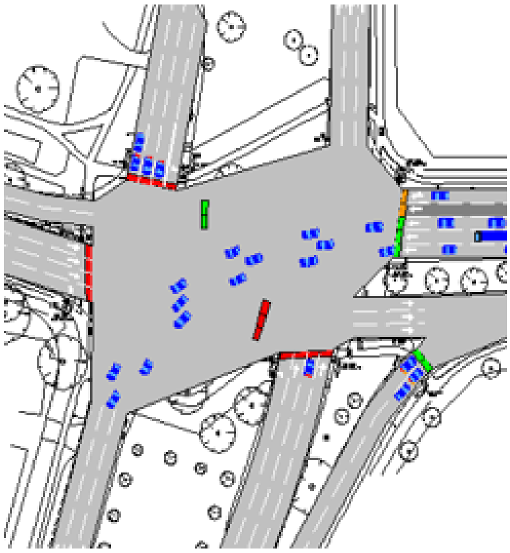

Figure 5: Example of an intersection model with Aimsun. The real intersection is located in Hamburg, Germany. The model has been provided by local authorities and has been initially developed by traffic engineers. It also includes the real-world traffic data from a census and the actual traffic control strategies.

Based on using OTC, we can investigate how to integrate SFI concepts and the corresponding AT behaviour. This further required to setup simulation models reflecting taxi-demands and owner behaviour in combination with real-world topology models, disturbances, and customer models—accompanied by appropriate metrics to assess the SFI's success.

This opens perspectives for optional additional components such as a trust and reputation system that may be incorporated in the decision processes as well.

5. Related Work

In the following, we discuss related work for the SFI scenario and the required technology from the field of SASO systems that is applicable in the context of the previously described research challenges.

5.1. State-of-the-Art in Related SASO Fields

Autonomous self-coordination: Works from this field focus on resource allocation without external intervention for finding an ordering of requesters by optimising a certain goal (e.g., priorities, cost, or fairness). In the context of the SFI, we consider three classes of autonomous self-coordination: (i) centralised approaches, (ii) negotiation-based approaches, and (iii) emergent approaches.

a) Centralised approaches use algorithms such as leader election for choosing one specific node that acts on behalf of the group (see e.g., [43] for an overview of algorithms). Afterwards, the resource allocation or coordination problem is handled in a centralised manner with the leader deciding about the current strategy. There are further concepts for exchanging the leader if it does not act as expected, for example, measured in terms of fairness metrics. Examples include [32] or [44].

b) Negotiation-based approaches are used alternatively to a centralised solution due to reasons such as a single point of failure, exploitation of power, communication overhead, or a variety of attack vectors [45]. Especially in the context of multi-agent systems, solutions among a group of autonomously acting agents that are considered to be equal have been investigated. Several situations can occur, where agents may not agree, but still need to find a consensus. This helps to achieve overall system reliability in the presence of a number of disagreeing agents. In general, this is referred to as "consensus problem" [46]. Approaches to tackle this include protocols (e.g., the Terminating Reliable Broadcast protocol [47] or the Contract Net protocol [48]) or mechanisms such as auctions [49].

c) Emergent-based approaches avoid explicit coordination or management. In turn, the system is fully decentralised as agents act autonomously without using explicit coordination or negotiation techniques. For coordination purposes, this generally refers to simple scheduling schemes, for example, first-come-first-serve (see [50] for an overview). Alternative solutions include OC concepts (e.g., [51]).

However, none of the approaches considered so far explicitly aims at establishing a hybrid optimisation scheme that is robust to changing conditions and reacts on different time scales, while simultaneously respecting the autonomy of individual subsystems.

Hybrid SASO systems: From a software engineering perspective, two attempts of designing systems can be distinguished. While bottom-up attempts usually focus on more practical concepts, they suffer at a missing clear conceptual separation in the design. As a result, the developed mechanisms are very specific for the underlying tasks. In contrast, top-down approaches usually try to achieve clear responsibilities and encapsulation, rendering the approach more generally applicable. However, this easily results in solutions that are less efficient for a specific solution. Some work has been spent on software engineering implications when establishing self-adaptive behaviour: For instance, Sudeikat et al. [52] describe a design concept for resource flow systems, where especially the coordination tasks are of interest; however, it does not consider learning and optimisation aspects.

There are only a few contributions in literature, where both perspectives are combined in order to find a trade-off. One particular example is the goal-oriented holonic system design concept [53], where the strict hierarchical composition of a system is replaced by holonic system organisation with goals as the primary coupling point. In general, hybrid SASO architectures integrate elements that have different scopes w.r.t. the centrality of their responsibilities, which can be categorized as (i) layered structures[e.g. 15, 5], (ii) cascaded structures[e.g. 38, 54], and (iii) hybrid coordination patterns[e.g. 55].

Optimisation in SASO systems: In [22], we analysed the usage of optimisation techniques within SASO systems for adaptation planning. We identified 29 different techniques in 51 publications, including probabilistic, combinatorial, evolutionary, stochastic, mathematical, and meta-heuristic optimisation techniques. The list is exemplary; however, the approaches target optimisation in centralised systems with only minor attention to decentralised optimisation. Such centralised optimisation comes with the disadvantages of providing the risk of a single point of failure as well as a potential bottleneck. Further, it contradicts the nature of autonomous entities. Decentralised optimisation techniques as required in our targeted system domain can overcome these issues.

Autonomous learning in SASO systems: In [56], we compared applied learning techniques in SASO systems. Here, Reinforcement Learning (RL) has been identified as most prominent variant, mainly realised as simple learning tasks (e.g., using Q-Learning) or more sophisticated approaches (e.g., using Learning Classifier Systems (LCS) [57]). Alternatively, Multi-agent RL (MARL) is employed to solve problems in a distributed manner when centralised control becomes infeasible [58]. However, all these techniques do not combine centralised planning with autonomous learning behaviour of the individual subsystems—and consequently do not focus the trade-off between optimality and autonomy. For our context, ML techniques for the self-improvement of coordination decisions of the autonomous subsystems at runtime are highly relevant. Especially the "Extended Classifier System" (XCS) variant by Wilson [30] has been widely used for implementing self-adaptation with runtime learning capabilities[e.g. 59, 60].

Multi-agent reinforcement learning (MARL): Multi-agent reinforcement learning is a subfield of artificial intelligence and machine learning that studies a potentially large set of (intelligent) agents coexisting and potentially cooperating in a shared environment. These agents are often modelled as working collaboratively toward a joint final goal. The main aspect of collaboration is usually following inspiration from social structures in natural systems and their interaction, e.g. animal states such as ants or termites. A second major aspect of the field has its roots in game theory, especially if learners are assumed to be not fully cooperative. Technical examples can be found in applications such as urban and air traffic control [61], multi-robot coordination [62, 63], distributed sensing [64], and energy distribution [65].

In MARL, several attempts to the coordination of learning agents exist. This includes examples such as [66], where an algorithm for the learning of organisational roles has been presented. In particular, the agents in the underlying scenario are modelled as being heterogeneous and the approach optimises the mapping of gent capabilities and tasks. In contrast to the learning problem in this article, their capabilities are less important and the autonomous decisions of the agent will overrule the mapping.

An alternative has been presented by Kok et al. [67]. Their work makes use of so-called coordination graphs, which are then used to solve the global coordination problem locally (given that it is possible to decompose the global payoff function, i.e. to turn it into the sum of local payoff functions). In [68], an extension has been proposed that infers the coordination graph online, which results in a transition from independent learners to coordinated action selection. Besides restrictions as before and the assumption that all agents are always willing and able to cooperate., it is hardly applicable to our problem due to relying on inference rules that are hand-crafted and the discrete nature of the state variables.

A third example has been presented by De Hauwere et al. [69]. The authors investigated a solution for mazes with two robots. The basic idea is to apply a generalised learning automaton working on the distance to the other robot to learn how to avoid a collision. This means recognising states that need to be coordinated. Since the approach itself is generic, it might be applicable to parts of the learning problem in this article, but the scope is too restricted in terms of the number of participants and requires manual intervention to adapt the inputs of the learning algorithm. There is some work that presented an extension towards several states, where still a selection of states is necessary [70]. In the work by Vrancx et al. [71] an extension to Coordinating Q-Learning has been proposed. Here, the authors apply transfer learning concepts on a state basis. This means that again a focus is on selected states that require coordination – which is not feasible in advance in our scenario (due to too many possible states, churn, and potentially unknown participants and jobs). Further, the underlying agent-centred view to generalise over varying conditions is not appropriate due to the missing possibility to distinguish between the concrete other systems.

More theoretically, Lanctot et al. [72] proposed a MARL-based metric that they called "joint policy correlation", which make use of repetitions of the same scenario with changing random influences resulting in different strategies of the agents. Based on this data, a matrix-based comparison of the the average rewards of the agents among the different repetitions is calculated. Using a very restricted two-agent laser-tag scenario, the obtained values against the initial opponent and the other opponents are aggregated to create a measure, and this measure describes how far an agent has overfitted to its initial opponent. Although this is another scope compared to this article's learning problem, the metric may serve as an indicator to identify how much an agent overfits the behaviour of other agents.

Anytime learning: To investigate the existing literature regarding anytime learning in SASO systems, we studied the publications of eleven journals and conferences. We identified 55 approaches; 33 learn in a centralised fashion, 22 apply distributed learning techniques. The papers show a diverse set of used optimisation approaches. The most common ones include evolutionary (16 papers), reinforcement learning (12 papers), heuristics (four papers), greedy (three papers), and tabu search (two papers) techniques. As one focus for the SFI is to increase the robustness through flexible plans and a fast adaptation to changes, our proposed approach needs to rely on anytime learning, especially, to handle periodical and event data as well as adjust the optimisation technique. The choice of the most suitable technique is part of our future work.

Incentivisation: At the core of an exchange economy with asymmetric information and/or limited availability of entities, incentives are used as a mechanism to guarantee participation [73]. As for the SFI scenario, some participants have strong benefits while others have fewer benefits (maybe even face a loss in their utility function), incentivisation is an important topic. In literature, different approaches to counter the negative effects (such as decreased willingness to participate or unfairness) are known, including compensation based on (crypt-)money, trust values, scheduling priorities, or reputation (for corresponding overviews see, e.g., [74, 75]). A particular characteristic that is of special importance in the coordination of SFI is the concept of delayed compensation based on incentives. Examples for such a system include a memory of investments (e.g., realised as trust values [76]) and the subsequent utilisation of these incentives [77]. Based on our work on fairness in platooning [21], we plan to analyse existing concepts for establishing fairness and computational trust as part of our future work.

Uncertainty: SASO systems often act in imperfect, dynamic environments. Hence, they have to incorporate the factors for uncertainty within the decision-making process. In the SFI scenario, uncertainty inherently arises from the fact the orders of customers might change or changes in the environment (e.g., accidents or traffic congestion) might not be anticipated. The SASO literature provides several approaches to address the uncertainty, which we shortly summarise in the following. For taming uncertainty, different requirement modelling languages and modelling approaches exist, for example FLAGS [78], CARE [79], RELAX [80], *LOREM* [81] or [82]. Another research stream focuses on the reasoning of adaptation under uncertainty. *POISED* [83] supports reasoning on uncertainty for the adaptation decision by evaluating the consequences of uncertainty using possibility theory; Moreno *et al.* [84] focuses on applying Markov decision processes for that purpose. Recently, Moreno *et al.* presented tactics to reduce uncertainty coming from simplified design assumptions, noise, model drift, context issues, human-in-the-loop, or decentralization [85]. Gerostathopoulos *et al.* present an approach to handle uncertainty resulting from noisy system outputs using Bayesian Optimisation with Gaussian Processes [86]. Kinneer proposes an approach for planning in unexpected situations by using prior planning knowledge based on genetic programming and reusing existing plans [87]. SimCA* provides a control-theoretic approach that provides guarantees for uncertainty related to system parameters, component interactions, system requirements, and environmental uncertainty [88]. Additionally, some authors focus on monitoring under uncertainty, [e.g. 89, 90].

5.2. State-of-the-Art in Fields Relevant for the SFI Technology

Current developments in autonomous driving and successful platforms for mobility services resulted in the prediction that—within the next decade—travellers will make increasing use of autonomous taxi services (also called 'robo-taxis'). Main drives for such a development are ubiquitous accessibility, easy utilisation and affordability [91, 92]. Several research efforts have been dedicated to the subsequent implications and business models for operating a fleet of such autonomous taxis [92]. The underlying idea is in most cases to establish a third-party organisation to respond to the travel demand of the entire urban population or a community (such as the SFI proposed above). The main utilisation pattern of such shared autonomous vehicles (SAVs) is assumed to be done by a closed group of members or centrally organised by a company for their employees, which in both cases neglects openness of the system and requires full pre-subscription.

Corresponding case studies and concepts for business models based on SAVs are, for instance, driven by large car manufacturers. Some of them try to establish themselves as the operator of the underlying infrastructure with the idea to earn money for provided services on a per kilometre or per trip basis [93]. The difference to the work presented in this article lies in a very abstract simulation environment and the limited scope, i.e., the focus on fleets operated by single vendors.

Current work mainly focuses on the design of the fleets and the operator, i.e., to establish the underlying basic operational characteristics for an upstream planning process. This includes aspects such as the fleet size, fleet specifications, relocation strategies, and the service area of the fleet [94].

A second interesting aspect of research on SFI-related topics is potential acceptance. Recently, we have seen several experimental studies on the acceptance of using AVs in general (see, e.g., [95]). Of particular interest here are those studies that analyse possible acceptability of SAVs. Examples include [96, 97] and [98]. The authors of these studies tried to estimate a possible market penetration rate based on the preferences queried by possible users. These are mostly static preferences without taking traffic conditions or dynamic alternatives into consideration. Due to the preliminary and abstract character of the AT service (since it is an envisioned future and not an available system), users tended to be sceptic in the first place. Consequently, researchers focused on first attempts to simulate prototypical behaviour of such systems to gain first insights. The most prominent approach is to rely on so-called "activity-based" multi-agent simulations, see [99] as an example. Based on a similar setting, [12] presented an extension to these simulations that are more interactive and demand-oriented. An alternative solution has been proposed in [100]. Here, the authors focus on investigations of the previously mentioned aspects of fleet size and optimisation strategies. Based on a case study in Austin, Texas, [101] increased the decision freedom of forming fleets.

All the studies and experiments mentioned in the previous paragraph rely on simulations using MATSim [39]. This is a major limitation since the abstractions and generalisations used in MATSim model the real-world at a macroscopic level. In particular, all real-world aspects and effects (e.g., real topology, complex phase systems, car follow and lane change models, or real-world traffic demands with realistic generation patterns) simulated by sophisticated traffic simulator such as Aimsun [42] are neglected.

Although this renders the transferability of the insights limited, recent developments show that the impact of user preferences on SAV usage can be beneficial. For instance, [102] showed—again based on MATSim simulations—that incorporating user preferences in scoring functions for SAV operation can serve as a basis for optimisation routines. In a particular case, the authors applied a co-evolutionary approach to optimise agent plans. In order to provide a realistic basis for the scenario, existing taxis are replaced by SAVs, and the service is assumed to be ubiquitously available (as a replacement). The simulation is limited again since it neglects the impact of waiting times. As an alternative, [103] modelled socio-demographic attributes such as age and income as key drivers for a discrete selection approach between available modes of transport, including SAVs. Since their simulation models real trip-taking activity—meaning all available modes at the time of a request—this is neglected for SAV mode (instead of using the conventional taxi mode attributes as in the study before).

A closely related field of research is the coordination of platooning, which is described as a cooperative driving technology where vehicles that are (partially) automated drive in close formation with gaps of three to ten meters [104]. Drivers do not have to control the forming of platoons as this is done automatically. The decision which platoon a vehicle should join relies on various parameters, for example, vehicle characteristics, planned velocity, or for how long the vehicles of a platoon can travel together, i.e., the overlap in their routes. Recently, researchers begin increasingly to work on the efficient assignment of vehicles to platoons; however, obeying individual constraints of drivers have been left out of scope. In [105], we present our concept for a platooning coordination approach. In contrast to existing approaches (see the overview in [106]), we integrate individual preferences of drivers, focus on multi-objective solutions, and provide individualised decisions which platoon to join. As platooning coordination represents a multi-level optimisation problem with a many- (different objectives from different vehicles) and multi-objective (integration of various global objectives) solution space, we think that approaches from this area might be transferred to the SFI.

Summarising the state-of-the-art in the field, we can state the following: Since autonomous driving is not ubiquitously available by now, research focused on simulations. By now, only abstract (mostly MATSim-based) studies are available. Studies regarding the preferences and the possible acceptability of SAV taxi services show that there is a possible market. However, the focus is on fleets operated by large companies (following the current conventional taxi model). A sharing model of private owners as envisioned in this article has not been investigated so far. However, insights into the current studies can be re-used in experiments.

6. Conclusion

This article introduced the SFI as a hybrid planning and optimisation framework for establishing autonomous taxi services in urban areas based on private-owned dynamically available autonomous cars. We explained that neither a fully centralised nor a fully distributed decision process will result in stable, optimised, and reliable behaviour. Consequently, we presented our vision of a system that combines abstract planning with decision freedoms respecting the autonomy of the individual participants—this system controls the behaviour by steering cost and compensation schemes that are effected as response to state estimations combining current observations and uncertainty-enriched forecasts of upcoming developments.

We proposed several research directions necessary to investigate the SFI system, combining efforts of four directions: a) anytime optimisation heuristics responsible for planning generalised role schemes with decision freedom of how to customise them as centralised dynamic service that continuously updates the schemes and steers the behaviour using commission fees and compensation payoffs, b) autonomous local learning techniques that continuously self-improve the behaviour of an individual car, c) awareness techniques for forecasting system states and quantifying the inherent uncertainty as well as an anomaly or novelty detection to trigger system and behaviour re-organisation, and d) simulation-based testbed that reflects real-world characteristics of the traffic conditions and dynamics. Our current and future work focuses on developing novel techniques and solutions to handle these challenges.

References

[1] E. Musk, The tesla vision, Online: http://www.cbc.ca/news/business/self-driving-tesla-expected-within-3-years-elon-musk-says-1.3107475 (last access: 19/05/2020) (2020).

[2] R. Vinuesa, H. Azizpour, I. Leite, M. Balaam, V. Dignum, S. Domisch, A. Felländer, S. D. Langhans, M. Tegmark, F. F. Nerini, The role of artificial intelligence in achieving the sustainable development goals, Nature communications 11 (1) (2020) 1–10.

[3] C. Müller-Schloer, S. Tomforde, Organic Computing – Techncial Systems for Survival in the Real World, Birkhäuser Verlag, 2017.

[4] J. Kephart, D. Chess, The Vision of Autonomic Computing, IEEE Computer 36 (1) (2003) 41–50.

[5] Samuel Kounev et al., The Notion of Self-aware Computing, in: Self-Aware Computing Systems, Springer, 2017, pp. 3–16.

[6] K. L. Bellman, S. Tomforde, R. P. Würtz, Interwoven systems: Self-improving systems integration, in: Eighth IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, SASOW 2014, London, United Kingdom, September 8-12, 2014, 2014, pp. 123–127.

[7] P. Angelov, Autonomous learning systems: From datastreamstoknowledgeinrealtime (2012).

[8] C. Krupitzer, F. M. Roth, S. VanSyckel, G. Schiele, C. Becker, A Survey on Engineering Approaches for Self-Adaptive Systems, PMCJ 17 (Part B) (2015) 184–206.

[9] S. Kounev, X. Zhu, J. O. Kephart, M. Kwiatkowska, Model-driven Algorithms and Architectures for Self-Aware Computing Systems (Dagstuhl Seminar 15041), Dagstuhl Reports 5 (1) (2015) 164–196. `doi:10.4230/DagRep.5.1.164`.
URL `http://drops.dagstuhl.de/opus/volltexte/2015/5038`

[10] V. Lesch, C. Krupitzer, S. Tomforde, Emerging self-integration through coordination of autonomous adaptive systems, in: Proc. FAS*W, 2019, pp. 6–9.

[11] S. Tomforde, C. Gruhl, B. Sick, A swarm-fleet infrastructure as a scenario for proactive, hybrid adaptation of system behaviour, in: Autonomic Computing and Self-Organizing Systems Companion (ACSOS-C), IEEE, 2020, pp. 166–169.

[12] S. Hörl, Agent-based simulation of autonomous taxi services with dynamic demand responses, Procedia Computer Science 109 (2017) 899–904.

[13] M. Maciejewski, J. Bischoff, Congestion effects of autonomous taxi fleets, Tailor and Francs Transport – Special Issue on Multi-Stakeholder Collaboration in Urban Transport (2016) 1–10.

[14] N. Criado, Using norms to control open multi-agent systems, AI Communications 26 (3) (2013) 317–318.

[15] S. Tomforde, H. Prothmann, J. Branke, J. Hähner, M. Mnif, C. Müller-Schloer, U. Richter, H. Schmeck, Observation and Control of Organic Systems, in: Organic Computing - A Paradigm Shift for Complex Systems, Birkhäuser, 2011, pp. 325 – 338.

[16] J. Hähner, S. von Mammen, S. Timpf, S. Tomforde, B. Sick, K. Geihs, T. Goeble, G. Hornung, G. Stumme, " know thyselves"-computational self-reflection in collective technical systems, in: Proc. of 29th Int. Conf. on Architecture of Computing Systems, 2016, pp. 1–8.

[17] S. Tomforde, C. Müller-Schloer, Incremental Design of Adaptive Systems, J. of Ambient Intelligence and Smart Environments 6 (2014) 179 – 198.

[18] M. Sommer, S. Tomforde, J. Hähner, An Organic Computing Approach to Resilient Traffic Management, in: Autonomic Road Transport Support Systems, Birkhäuser, 2016, pp. 113–130.

[19] A. Stein, S. Rudolph, S. Tomforde, J. Hähner, Self-learning smart cameras - harnessing the generalization capability of XCS, in: Proceedings of the 9th International Joint Conference on Computational Intelligence, IJCCI 2017, Funchal, Madeira, Portugal, November 1-3, 2017, 2017, pp. 129–140.

[20] V. Lesch, C. Krupitzer, S. Tomforde, Multi-objective optimisation in hybrid collaborating adaptive systems, in: Proc. SAOS, 2019.

[21] V. Lesch, C. Krupitzer, K. Stubenrauch, N. Keil, C. Becker, S. Kounev, M. Segata, A comparison of mechanisms for compensating negative impacts of system integration, Future Generation Computer Systems 116 (2021) 117 – 131. `doi:https://doi.org/10.1016/j.future.2020.10.029`.

[22] E. M. Fredericks, I. Gerostathopoulos, C. Krupitzer, T. Vogel, Planning as optimization: Dynamically discovering optimal configurations for runtime situations, in: Proc. SASO, 2019.

[23] C. Gruhl, J. Schmeißing, S. Tomforde, B. Sick, Normal-wishart clustering for novelty detection, in: Autonomic Computing and Self-Organizing Systems Companion (ACSOS-C), IEEE, 2020, pp. 64–69.

[24] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, A. Bouchachia, A survey on concept drift adaptation, ACM computing surveys 46 (4) (2014) 1–37.

[25] E. R. Faria, I. J. Gonçalves, A. C. de Carvalho, J. Gama, Novelty detection in data streams, Art. Int. Rev. 45(2) (2016) 235–269.

[26] T. Gneiting, M. Katzfuss, Probabilistic forecasting, Annual Review of Statistics and Its Application 1 (2014) 125–151.

[27] D. Salinas, V. Flunkert, J. Gasthaus, T. Januschowski, Deepar: Probabilistic forecasting with autoregressive recurrent networks, International Journal of Forecasting (2019).

[28] W. Zhang, H. Quan, D. Srinivasan, An improved quantile regression neural network for probabilistic load forecasting, IEEE Transactions on Smart Grid 10 (4) (2018) 4425–4434.

[29] H. Wang, and others, Deterministic and probabilistic forecasting of photovoltaic power based on deep convolutional neural network, Energy conversion and management 153 (2017) 409–422.

[30] S. W. Wilson, Classifier Fitness Based on Accuracy, Evolutionary Computation 3 (2) (1995) 149–175.

[31] G. D. Howard, L. Bull, P.-L. Lanzi, Towards continuous actions in continuous space and time using self-adaptive constructivism in neural xcsf, in: Proc. of 11th conf. on Genetic and Evolutionary Comp., 2009, pp. 1219–1226.

[32] S. Edenhofer, S. Tomforde, J. Kantert, L. Klejnowski, Y. Bernard, J. Hähner, C. Müller-Schloer, Trust communities: An open, self-organised social infrastructure of autonomous agents, in: Trustworthy Open Self-Organising Systems, 2016, pp. 127–152.

[33] A. Calma, D. Kottke, B. Sick, S. Tomforde, Learning to learn: Dynamic

runtime exploitation of various knowledge sources and machine learning paradigms, in: 2nd IEEE International Workshops on Foundations and Applications of Self* Systems, FAS*W@SASO/ICCAC 2017, Tucson, AZ, USA, September 18-22, 2017, 2017, pp. 109–116.

[34] L. Busoniu, R. Babuska, B. De Schutter, A comprehensive survey of multiagent reinforcement learning, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 38 (2) (2008) 156–172.

[35] P. Stone, M. Veloso, Multiagent systems: A survey from a machine learning perspective, Autonomous Robots 8 (3) (2000) 345–383.

[36] S. Rudolph, S. Tomforde, J. Hähner, Mutual Influence-Aware Runtime Learning of Self-Adaptation Behaviour, ACM TAAS 14 (1) (2019).

[37] S. VanSyckel, D. Schäfer, V. Majuntke, C. Krupitzer, G. Schiele, C. Becker, Comity: A framework for adaptation coordination in multi-platform pervasive systems, Pervasive and Mobile Computing 10 (2014) 51–65.

[38] J. O. Kephart, D. M. Chess, The Vision of Autonomic Computing, IEEE Computer 36 (1) (2003) 41–50.

[39] K. W Axhausen, A. Horni, K. Nagel, The multi-agent transport simulation MATSim, Ubiquity Press, 2016.

[40] S. Tomforde, H. Prothmann, F. Rochner, J. Branke, J. Hähner, C. Müller-Schloer, H. Schmeck, Decentralised Progressive Signal Systems for Organic Traffic Control, in: Proc. SASO, 2008, pp. 413–422.

[41] H. Prothmann, S. Tomforde, J. Lyda, J. Branke, J. Hähner, C. Müller-Schloer, H. Schmeck, Self-organised routing for road networks, in: International Workshop on Self-Organizing Systems, Springer, 2012, pp. 48–59.

[42] J. Casas, J. L. Ferrer, D. Garcia, J. Perarnau, A. Torday, Traffic simulation with aimsun, in: Fundamentals of traffic simulation, Springer, 2010, pp. 173–232.

[43] S. Dolev, A. Israeli, S. Moran, Uniform dynamic self-stabilizing leader election, IEEE TPDS 8 (4) (1997) 424–440.

[44] M. Ji, A. Muhammad, M. Egerstedt, Leader-based multi-agent coordination: Controllability and optimal control, in: Proc. ACC, 2006.

[45] M. Weißbach, P. Chrszon, T. Springer, A. Schill, Decentrally coordinated execution of adaptations in distributed self-adaptive software systems, in: Proc. SASO, 2017, pp. 111–120.

[46] G. F. Coulouris, J. Dollimore, T. Kindberg, Distributed systems: concepts and design, pearson education, 2005.

[47] G. Bracha, Asynchronous byzantine agreement protocols, Information and Computation 75 (2) (1987) 130–143.

[48] R. G. Smith, The contract net protocol: High-level communication and control in a distributed problem solver, IEEE Trans. Computers (12) (1980) 1104–1113.

[49] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, M. J. Wooldridge, C. Sierra, Automated negotiation: prospects, methods and challenges, Group Decision and Negotiation 10 (2) (2001) 199–215.

[50] M. L. Pinedo, Scheduling: theory, algorithms, and systems, Springer, 2016.

[51] A. Scheidler, D. Merkle, M. Middendorf, Congestion control in ant like moving agent systems, in: Proc. Biologically-Inspired Collaborative Computing, 2008, pp. 33–43.

[52] J. Sudeikat, L. Braubach, A. Pokahr, W. Renz, W. Lamersdorf, Systematically engineering self-organizing systems: The sodekovs approach, Electronic Communications of the EASST 17 (2009).

[53] A. Diaconescu, S. Frey, C. Müller-Schloer, J. Pitt, S. Tomforde, Goal-oriented Holonics for Complex System (Self-)Integration: Concepts and Case Studies, in: Proc. SASO, 2016.

[54] T. Vogel, H. Giese, Model-Driven Engineering of Self-Adaptive Software with EUREMA, ACM TAAS 8 (4) (2014) 18:1–18:33.

[55] D. Weyns, B. R. Schmerl, V. Grassi, S. Malek, R. Mirandola, C. Prehofer, J. Wuttke, J. Andersson, H. Giese, K. M. Göschka, On Patterns for Decentralized Control in Self-Adaptive Systems, in: Software Engineering for Self-Adaptive Systems II, Springer, 2013, pp. 76–107.

[56] M. D'Angelo, S. Gerasimou, S. Ghahremani, J. Grohmann, I. Nunes, E. Pournaras, S. Tomforde, On learning in collective self-adaptive systems: state of practice and a 3d framework, in: Proc. SEAMS, 2019.

[57] Y. Rizk, M. Awad, E. Tunstel, Decision making in multiagent systems: a survey, IEEE TCDS 10 (3) (2018) 514–529.

[58] A. Marinescu, I. Dusparic, S. Clarke, Prediction-based multi-agent reinforcement learning in inherently non-stationary environments, ACM TAAS 12 (2) (2017) 9:1–9:23.

[59] A. Stein, D. Rauh, S. Tomforde, J. Hähner, Interpolation in the eXtended Classifier System: An Architectural Perspective, Journal of Systems Architecture (2017).

[60] S. Tomforde, J. Hähner, Biologically Inspired Networking and Sensing: Algorithms and Architectures, IGI, 2011, Ch. Organic Network Control – Turning Standard Protocols Into Evolving Systems, pp. 11 – 35.

[61] A. K. Agogino, K. Tumer, A multiagent approach to managing air traffic flow, Autonomous Agents and Multi-Agent Systems 24 (1) (2012) 1–25.

[62] M. Ahmadi, P. Stone, A multi-robot system for continuous area sweeping tasks, in: Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006., IEEE, 2006, pp. 1724–1729.

[63] D. Hennes, D. Claes, W. Meeussen, K. Tuyls, Multi-robot collision avoidance with localization uncertainty., in: AAMAS, 2012, pp. 147–154.

[64] M. Mihaylov, K. Tuyls, A. Nowé, A decentralized approach for convention emergence in multi-agent systems, Autonomous Agents and Multi-Agent Systems 28 (5) (2014) 749–778.

[65] M. Pipattanasomporn, H. Feroze, S. Rahman, Multi-agent systems in a distributed smart grid: Design and implementation, in: 2009 IEEE/PES Power Systems Conference and Exposition, IEEE, 2009, pp. 1–8.

[66] M. N. Prasad, V. R. Lesser, S. E. Lander, Learning organizational roles in a heterogeneous multi-agent system, in: Adaptation, Coevolution and Learning in Multiagent Systems: Papers from the 1996 AAAI Spring Symposium, 1996, pp. 72–77.

[67] J. R. Kok, M. T. Spaan, N. Vlassis, et al., Multi-robot decision making using coordination graphs, in: Proceedings of the 11th International Conference on Advanced Robotics, ICAR, Vol. 3, 2003, pp. 1124–1129.

[68] J. R. Kok, E. J. Hoen, B. Bakker, N. Vlassis, Utile coordination: Learning interdependencies among cooperative agents, in: EEE Symp. on Computational Intelligence and Games, Colchester, Essex, 2005, pp. 29–36.

[69] Y.-M. De Hauwere, P. Vrancx, A. Nowé, Learning what to observe in multi-agent systems, in: Proceedings of the 20th Belgian-Netherlands Conference on Artificial Intelligence, 2009, pp. 83–90.

[70] Y.-M. De Hauwere, P. Vrancx, A. Nowé, Learning multi-agent state space representations, in: Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1, 2010, pp. 715–722.

[71] P. Vrancx, Y.-M. De Hauwere, A. Nowé, Transfer learning for multi-agent coordination., in: ICAART (2), Rome, Italy, 2011, pp. 263–272.

[72] M. Lanctot, V. Zambaldi, A. Gruslys, A. Lazaridou, K. Tuyls, J. Pérolat, D. Silver, T. Graepel, A unified game-theoretic approach to multiagent reinforcement learning, arXiv preprint arXiv:1711.00832 (2017).

[73] F. Forges, E. Minelli, R. Vohra, Incentives and the core of an exchange economy: a survey, Journal of Mathematical Economics 38 (1-2) (2002) 1–41.

[74] T. Requate, Dynamic incentives by environmental policy instruments—a survey, Ecological economics 54 (2-3) (2005) 175–195.

[75] X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, X. Mao, Incentives for mobile crowd sensing: A survey, IEEE Communications Surveys & Tutorials 18 (1) (2015) 54–67.

[76] J. Kantert, S. Edenhofer, S. Tomforde, C. Müller-Schloer, Representation of Trust and Reputation in Self-Managed Computing Systems, in: Proc. DASC, 2015.

[77] J. Kantert, S. Tomforde, A. Diaconescu, C. Müller-Schloer, Incentive-oriented Task Assignment in Holonic Organic Systems, in: Proc. SAOS, 2017.

[78] L. Baresi, L. Pasquale, P. Spoletini, Fuzzy Goals for Requirements-Driven Adaptation, in: Proc. RE, 2010, pp. 125–134.

[79] N. A. Qureshi, I. J. Jureta, A. Perini, Towards a Requirements Modeling Language for Self-Adaptive Systems, in: Requirements Engineering: Foundation for Software Quality, Springer, 2012, pp. 263–279.

[80] J. Whittle, P. Sawyer, N. Bencomo, B. H. C. Cheng, J.-M. Bruel, RELAX: a language to address uncertainty in self-adaptive systems requirement, Requirements Engineering 15 (2) (2010).

[81] D. M. Berry, B. H. C. Cheng, J. Zhang, The Four Levels of Requirements Engineering for and in Dynamic Adaptive Systems, in: Proc. REFSQ, 2005.

[82] A. J. Ramirez, B. H. C. Cheng, N. Bencomo, P. Sawyer, Relaxing

Claims: Coping with Uncertainty While Evaluating Assumptions at Run Time, in: Model Driven Engineering Languages and Systems, Springer, 2012, pp. 53–69.
[83] N. Esfahani, E. Kouroshfar, S. Malek, Taming uncertainty in self-adaptive software, in: Proc. ESEC/FSE, 2011.
[84] G. A. Moreno, J. Cámara, D. Garlan, B. Schmerl, Efficient Decision-Making under Uncertainty for Proactive Self-Adaptation, in: Proc. ICAC, 2016, pp. 147–156.
[85] G. A. Moreno, J. Cámara, D. Garlan, M. Klein, Uncertainty Reduction in Self-Adaptive Systems, in: Proc. SEAMS, 2018, pp. 51–57.
[86] I. Gerostathopoulos, C. Prehofer, T. Bures, Adapting a System with Noisy Outputs with Statistical Guarantees, in: Proc. SEAMS, 2018.
[87] C. Kinneer, Z. Coker, J. Wang, D. Garlan, C. L. Goues, Managing Uncertainty in Self-Adaptive Systems with Plan Reuse and Stochastic Search, in: Proc. SEAMS, 2018, pp. 40–50.
[88] S. Shevtsov, D. Weyns, M. Maggio, SimCA*: A Control-Theoretic Approach to Handle Uncertainty in Self-Adaptive Systems with Guarantees, ACM TAAS 13 (4) (2019).
[89] P. Casanova, D. Garlan, B. Schmerl, R. Abreu, Diagnosing Unobserved Components in Self-adaptive Systems, in: Proc. SEAMS, 2014, pp. 75–84.
[90] G. Weiss, P. Schleiss, D. Schneider, M. Trapp, Towards Integrating Undependable Self-Adaptive Systems in Safety-Critical Environments, in: Proc. SEAMS, 2018, pp. 26–32.
[91] N. D. Chan, S. A. Shaheen, Ridesharing in north america: Past, present, and future, Transport reviews 32 (1) (2012) 93–112.
[92] S. A. Shaheen, N. D. Chan, H. Micheaux, One-way carsharing's evolution and operator perspectives from the americas, Transportation 42 (3) (2015) 519–536.
[93] J. Firnkorn, M. Müller, Selling mobility instead of cars: new business strategies of automakers and the impact on private vehicle holding, Business Strategy and the environment 21 (4) (2012) 264–280.
[94] R. Vosooghi, J. Kamel, J. Puchinger, V. Leblond, M. Jankovic, Robo-taxi service fleet sizing: assessing the impact of user trust and willingness-to-use, Transportation 46 (6) (2019) 1997–2015.
[95] C. J. Haboucha, R. Ishaq, Y. Shiftan, User preferences regarding autonomous vehicles, Transportation Research Part C: Emerging Technologies 78 (2017) 37–49.
[96] P. Bansal, K. M. Kockelman, A. Singh, Assessing public opinions of and interest in new vehicle technologies: An austin perspective, Transportation Research Part C: Emerging Technologies 67 (2016) 1–14.
[97] R. Krueger, T. H. Rashidi, J. M. Rose, Preferences for shared autonomous vehicles, Transportation research part C: emerging technologies 69 (2016) 343–355.
[98] F. Steck, V. Kolarova, F. Bahamonde-Birke, S. Trommer, B. Lenz, How autonomous driving may affect the value of travel time savings for commuting, Transportation research record 2672 (46) (2018) 11–20.
[99] R. Vosooghi, J. Puchinger, M. Jankovic, G. Sirin, A critical analysis of travel demand estimation for new one-way carsharing systems, in: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2017, pp. 199–205.
[100] B. Wang, S. A. O. Medina, P. Fourie, Simulation of autonomous transit on demand for fleet size and deployment strategy optimization, Procedia computer science 130 (2018) 797–802.
[101] D. J. Fagnant, K. M. Kockelman, Dynamic ride-sharing and fleet sizing for a system of shared autonomous vehicles in austin, texas, Transportation 45 (1) (2018) 143–158.
[102] J. Kamel, R. Vosooghi, J. Puchinger, F. Ksontini, G. Sirin, Exploring the impact of user preferences on shared autonomous vehicle modal split: A multi-agent simulation approach, Transportation Research Procedia 37 (2019) 115–122.
[103] L. M. Martinez, J. M. Viegas, Assessing the impacts of deploying a shared self-driving urban mobility system: An agent-based model applied to the city of lisbon, portugal, International Journal of Transportation Science and Technology 6 (1) (2017) 13–27.
[104] C. Bergenhem, H. Petterson, E. Coelingh, C. Englund, S. Shladover, S. Tsugawa, Overview of Platooning Systems, in: Proc. ITSWC, 2012, pp. 1393–1407.
[105] C. Krupitzer, M. Segata, M. Breitbach, S. El-Tawab, S. Tomforde, C. Becker, Towards Infrastructure-Aided Self-Organized hybrid platooning, in: Proc. GCIoT, 2018, pp. 1–6.
[106] A. K. Bhoopalam, N. Agatz, R. Zuidwijk, Planning of truck platoons: A literature review and directions for future research, Transportation Research Part B: Methodological 107 (2018) 212–228.