Utility-based Vehicle Routing Integrating User Preferences

Veronika Lesch, Marius Hadry, Samuel Kounev
Department of Computer Science
University of Würzburg
Würzburg, Germany
{firstname.lastname}@uni-wuerzburg.de

Christian Krupitzer
Department of Food Informatics & Comp. Science Lab
University of Hohenheim
Stuttgart, Germany
christian.krupitzer@uni-hohenheim.de

***Abstract*—To fulfill Mark Weiser's vision that "the most profound technologies are those that disappear", pervasive computing systems inherently have (i) to interact with the environment and the users but also (ii) to adapt their behavior to changes in their environment. Due to the dynamic, non-deterministic environment of pervasive computing systems, uncertainty arises. In this paper, we conduct a case study in the domain of driver assistance systems and show that uncertainty in proactive adaptations, i.e., adaptations in advance, increases with an extension of the planning horizon.**

***Index Terms*—Uncertainty, human-in-the-loop, self-adaptive systems, utility function, vehicle routing**

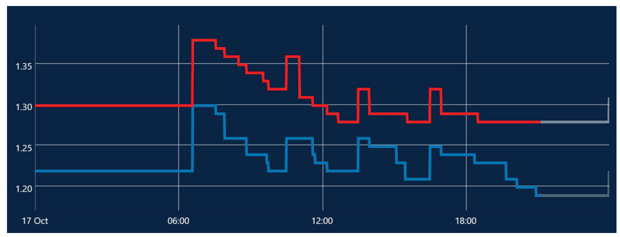

Fig. 1. Fuel price of a gas station in Heidelberg, Germany on Oct., 17, 2019 (data from www.clever-tanken.de).

I. Introduction

Pervasive computing systems aim to provide seamless support of users in their daily life with intelligent information technology [1]. Accordingly, those systems have to automatically adjust their behavior to changes in their context as well as the user preferences or the user group, respectively. As an adaptation in reaction to those changes always includes a delay—and potentially downtime in the service of those systems—planning in advance and proactive adaptation might be beneficial. However, the larger the scope for the planning in advance, the higher the risk of unanticipated changes or wrong forecasts of future system and environment states due to uncertainty in the dynamic environment of the system. Human-in-the-loop integration as an essential requirement for interacting pervasive systems increases the severity of the uncertainty issue as users' behavior is uncertain and, further, the users' objectives are difficult to predict. However, those highly influence the process of adaptation planning [2].

In this paper, we study the effects of uncertainty in the scope of proactive adaptation planning. This means that we already take the effects of uncertainty in the planning phase into account to minimize the probability of unforeseen change affecting the adaptation. Based on the observation of highly volatile fuel prices in some countries (see an example from Germany in Figure 1), we study those claims in CostSAVeR, a pervasive navigation system that applies multi-criteria optimization for cost-aware routing. The application includes a decision logic to analyze possible routes using utility functions. Using our evaluation testbed, we assess the performance of our utility functions as well as the importance of reasoning on the uncertainty of price stability with travel distance or time. We disregard an ad-hoc approach that continually analyzes the route. This shifts the uncertainty in the adaptation to another point of time as it is never guaranteed that the currently cost-optimal route will finally be the cost-optimal one. Summarizing, we make the following contributions:

- Definition of utility functions that support cost-aware decision making under uncertainty in our application area.
- Comparison of our cost-aware utility functions using a self-developed, reusable evaluation testbed.

The remainder of this paper is structured as follows. First, we summarize related work and discuss the limitations of these approaches in Section II. Next, Section III presents the use case system for the case study and six different utility functions that cover aspects of cost-aware vehicle routing under uncertainty. Section IV describes the results of our case study investigating the influence of uncertainty for proactive adaptation planning in different planning horizons as well as the influence of user preferences on the adaptation planning. Based on the case study, Finally, Section V summarizes the paper and gives an outlook into future work.

II. Related Work

In pervasive computing systems, often more than one attribute influences the decision-making process. In literature, several authors propose a multi-dimensional approach for utility functions [3], [4]. Thus, each adaptation dimension has its own independent utility function. The adaptation is decided based on the weighted sum of all utility values [4]. So far, research focuses more on incorporating different factors into a utility function rather than choosing or switching between utility functions at runtime. With this work, we want to motivate why this can be beneficial.

Human-in-the-loop interaction is often seen as a potential source of uncertainty [2], [5]. Cámara *et al.* discuss different types of involving humans in self-adaptive systems but focus on integrating humans for doing tasks that are difficult or infeasible to automate [6]. Here, the human is seen in the function of a system administrator. Similarly, the framework of Gil *et al.* provides a design of human participation in the control loops [7]. Huang and Miranda present an approach to adding users' intentions through neural input into the adaptation decision [8]. Consequently, this enables to adjust the system's behavior to the goals of the users. Similarly, Becker, Hähner, and Tomforde present an approach to integrate flexibility through incorporating changing user goals in a learning-based adaptation decision making [9]. Cámara, Moreno, and Garlan define a modeling approach for reasoning about the humans' capability for being involved in self-adaptation. The modeling approach relies on stochastic multiplayer games [10]. Summarizing, those works integrate humans for (i) administrative tasks, (ii) reasoning on system goals, and (iii) incorporating information about the user's context to optimize the service. However, little work is done on investigating how to switch between different utility functions depending on the users' preferences. In this work, we motivate the need for such a meta-adaptation of the adaptation planning mechanism by comparing the appropriateness of different utility functions to measure various objectives.

III. Cost-aware, Adaptive Vehicle Routing

Several authors proposed adaptiveness for vehicle routing (e.g., [11], [12]) and traffic management (e.g., [13], [14]). In line, we also study the effects of uncertainty and integrating user preferences in the adaptation decision process in a case study on traffic navigation, the CostSAVeR system. As CostSAVeR operates in a highly dynamic environment in which (i) the fuel prices are highly volatile and (ii) the traffic conditions vary spontaneously due to accidents, construction works or variability in the traffic volume, we are faced with high uncertainty in the planning of trips. To handle those circumstances, we design our system as a self-adaptive system [15]. Hence, the system is able to modify its parameters and utility functions at run-time to adapt to the changes in its environment. In our case, we focus on an adjustment of the route for re-directing a vehicle (i) to a gas station and (ii) as a reaction to changing traffic conditions.

A. CostSAVeR

We designed our system as a self-adaptive system composed of a managing subsystem, called adaptation logic, which controls and adapts a managed subsystem [15]. The adaptation logic integrates a MAPE control loop [16] for controlling the adaptation. It incorporates the functionality for monitoring the environment – i.e., fuel prices, the current traffic flow, as well as possible traffic congestion – and the managed subsystems, for analyzing the situation, for planning the route, and for outputting the result to the UI or an interface for autonomous vehicles. Adaptations in our use case depend on the route, on user-specified optimization constraints, or the dynamics of the traffic circulation. We integrate cost-efficiency as such constraint, hence, adaptations can be caused by changes in current fuel prices. However, for future work, it might be possible to add other factors for personalized routing.

We implemented CostSAVeR as a web-based prototype. As frontend client, an Android application[1] supports our adaptive navigation. It further supports real-time navigation based on the Google Maps navigation service. The backend receives the requests from the frontend, calculates the utility functions, and returns a ranked list of alternative routes to the frontend. In case of the web frontend, the calculation is performed once before the start of the journey. The Android app is able to adjust the route while driving. As the backend delivers a set of possible assessed routes so that the user can decide according to his preferences (e.g., using a specific route or focusing on gas station brands) our approach integrates the users in the loop.

First, the required input is collected from the connected interfaces[2]: preference of the user (i.e., the goal of the user), origin and destination of the trip, average fuel consumption, price of the last filling, vehicle type, fuel type, and remaining driving range. Besides the user-related data, the system collects data about its environment, that is, alternative routes, traffic information, and fuel prices. Here, we use the *Google Maps API* for requesting routes, the Here WeGo API for retrieval of gas stations, and the Tankerkoenig API[3] for requesting current gas prices. Next, the system uses the collected data to identify the current situation and to analyze if

[1] We published the Android installation file anonymously on zenodo: https://doi.org/10.5281/zenodo.4067966.

[2] We focus on the collection of data from the user interface and omit the integration of the OBD-II interface to collect data from the car.

[3] https://www.tankerkoenig.de/

the current situation requires an update of the route. We analyze a form of proactive adaptation by planning refueling in advance, i.e., at the beginning of the trip, and omit run-time adaptation as a reaction to changes in the fuel prices. Afterwards, the system performs the calculation of the possible routes and the optimization by determining the quality of the routes. Therefore, the planning component uses the information retrieved in the monitoring step combined with the determined situation of the analyze step. The combination of information about routes and gas stations forms new routes with potential detours for reaching a gas station. Each of these routes is assessed using the later defined utility functions. The planning component returns a ranked list of routes. For choosing a route, the system either integrates the user or decides autonomously, for example, in the case of an autonomous vehicle.

B. Utility Functions

Utility functions represent one way to evaluate which adaptation from a search space fits best to perform self-optimization [17]. We introduce six different utility functions as a representative set of three categories: (i) integrating measures of the gas price, distance, and duration; (ii) coping with the uncertainty of volatile gas prices; and (iii) selecting either the nearest or completely random gas stations. Each utility function calculates the utility per route.

Price-aware. The first utility function integrates the already paid costs for the remaining amount of fuel and the new refueling costs. Equation (1) presents the calculation of the estimated costs using the price per liter of the last refueling (p_{last}), the number of liters remaining ($l_{remaining}$), the current price per liter (p_{cur}) at the desired station, and the number of liters required (l_{new}) for the planned tour.

$$cost_{est} = p_{last} \cdot l_{remaining} + p_{cur} \cdot l_{new} \tag{1}$$

For using the estimated costs as utility, we define a vector presentation of the estimated costs for all routes $cost_{est}[i]$ with i being the index of the i-th route. We use a modified version of the min-max scaling formula [18] to calculate a ranking. Thus, Equation (2) normalizes the values to the bounds [0, 1] so that the highest costs have a utility of 0 and the smallest costs a utility of 1:

$$x[i]_{reversed} = \frac{x[i] - \max(x)}{\min(x) - \max(x)} \tag{2}$$

By integrating the described cost calculation and the min-max scaling, the price-aware utility function is defined as:

$$U_{Pr} = \frac{cost_{est}[i] - \max(cost_{est})}{\min(cost_{est}) - \max(cost_{est})} \tag{3}$$

Duration-/Distance-aware. The *duration-/distance-aware* utility function (Dur/Dist) also uses min-max scaling and includes the duration and distance into one function by forming a weighted sum as defined in Equation 4. Again, the reversed min-max scaling of Equation (2) is used. The developer defines the weights w_i, which must sum up to one. Considered attributes i might be costs or duration.

$$U_{DD} = \sum_i w_i \cdot x[i]_{reversed} \tag{4}$$

Since the other utility functions mainly focus on the cost attribute, weighting every attribute equally in this work will show the other attributes' impact on the solutions. These weights can be changed in the future to get different solutions but are fixed for our work.

Volatility-aware. Since gas prices are highly volatile, the *volatility-aware* utility function tries to minimize the uncertainty of price changes. Based on the assumption that the probability of a change of the price at a gas station increases with a larger distance towards the gas station, the idea is to reward closer gas stations with a utility bonus. The price-aware utility function defined in Equation (5) is calculated and a bonus is added concerning the distance of the station ($d_{station}$) from the distance to the destination (d_{dest}). We introduce the weights are called α for weighting the utility of the price-aware utility function and β for weighting the added bonus.

$$U_V = U_{Pr} * \alpha + \left(1 - \frac{d_{station}}{d_{dest}}\right) * \beta \tag{5}$$

We set the parameter α to $\frac{3}{4}$ and the parameter β to $\frac{1}{4}$. The reason for this weighting is to still let the cost factor dominate the function. Otherwise, only gas stations that are very close to the origin might be chosen.

Penalty-aware. The *penalty-aware* utility function also tries to minimize unforeseen changes in prices. In contrast to the volatility-aware utility function, it uses the required time it takes to reach the gas station ($t_{station}$) instead of rewarding closer distances. A punishment p is added to the price-aware utility function for each period ($period$) it takes to reach the station as defined in Equation (6).

$$U_{Pen} = U_{Pr} - \left\lfloor \frac{t_{station}}{period} \right\rfloor * p \tag{6}$$

We set the $period$ to a value of 1800 seconds and the value of p to 0.05.

Nearest Station. With this utility function, we model behavior to refuel at the closest gas station. Therefore, we use the start location for a distance-based search for the closest gas station and calculate the costs according to Equation (1). In the evaluation, this utility function serves as a comparison for the other utility functions as it does not consider any route or cost-awareness.

Random. This utility function selects a gas station among the identified gas stations within the search radius

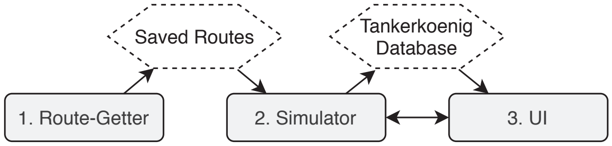

Fig. 2. Workflow of the evaluation tool showing the simulation core at the bottom center as well as different input sources and a user interface.

randomly. This process is performed 30 times to receive average costs for comparison.

IV. Evaluation

We implemented a tool that evaluates the quality and appropriateness of our utility functions w.r.t. user preferences as well as uncertainty in the planning horizon. To guarantee our results' reproducibility, we designed our tool to contain all the required data for our evaluation.

A. Evaluation Tool

Our tool evaluates the quality of the utility functions w.r.t. user preferences and uncertainty in the planning horizon. Therefore, the tool integrates all utility functions and limits the execution time of the evaluation runs by storing predefined routes and all gas stations alongside. We use *historical* gas prices provided by Tankerkoenig combined with predefined routes as we want to compare the actual gas prices once the driver arrives at the station. This allows for an in-depth evaluation of the utility functions with the advantage of being fully reproducible when the routes, found gas stations, and departure times are identical.

Figure 2 displays the workflow of used components and saved data of the evaluation tool. First, the *Route-Getter* provides automation of downloading relevant information for the predefined routes. Using the Google Maps API, the Route-Getter receives the routes between origin and destination. Afterward, the Here WeGo API is used to find possible gas stations within a specified radius and the Google Maps API is again used to find routes that integrate the gas stations as waypoints - one at a time. The Route-Getter component is used only once in an early phase of this work to retrieve data of a predefined set of routes and dates. The data is stored locally as *Saved Routes*. In the second step, the *Simulator* combines the saved data and implements the utility functions from Section III-B. The Tankerkoenig database provides locally stored historical fuel prices. Third, a frontend is provided, enabling the user to overview the evaluation results easily. This can be done in two ways: (i) evaluation of a single route at one predefined date or (ii) evaluation of all possible routes and a set of predefined dates automatically.

B. Metrics

As the gas prices might change during travel time to the gas station, we introduce two metrics and one additional utility function to evaluate our utility functions' quality.

Estimated Costs: The utility functions calculate the estimated costs based on the gas prices at planning time.

Actual Costs: The actual costs are the costs when using the actual fuel prices at arrival at the gas station for calculating the costs with the utility function.

Optimum Costs (OPT): The *OPT* utility function is a theoretical utility function that is used to calculate the utility functions' theoretical optimum when having comprehensive information regarding possibly changing gas prices. This utility function is calculated the same way as the price-aware utility function but with the arrival time's actual fuel prices. As it incorporates the actual gas prices, we can evaluate retrospectively which gas station would have been cost-optimal for this scenario.

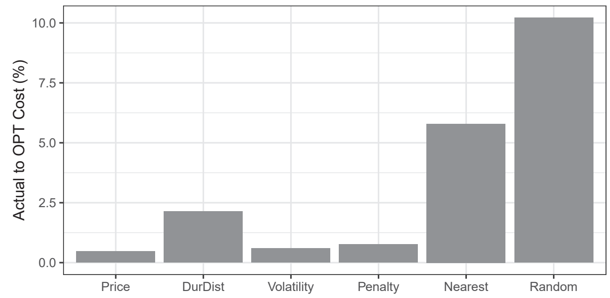

Fig. 3. Percentage deviation of actual and optimal costs.

C. Evaluation Methodology

To evaluate the performance of the different utility functions, we aimed at defining a representative data set concerning different routes all over Germany and several dates during the year 2018. The 22 selected routes cover Germany's roadside to a large extent and show differing route lengths (between 50 km and 900 km) and used road types. The 18 dates are chosen with regards to vacations or holidays, regular weekdays, and weekends. For each date, three different timestamps are used as the start time of the journey: 6:00 AM, 12:00 PM, and 4:00 PM. The average cost per route is around 25 €.

D. Quality of the Utility Functions

To evaluate the quality of the used utility functions, we look at the cost deviation the selected route has to the theoretical optimum (cf. Section IV-B). Figure 3 shows the mean deviation of the costs for the selected route from the optimum costs for each utility function in percent. The mean is calculated over all combinations of routes and dates.

The figure shows that price-aware, volatility-aware, and penalty-aware utility functions have the lowest percentage deviation to the optimum of around 0.4%, which refers to a mean loss of 0.09 € per route.

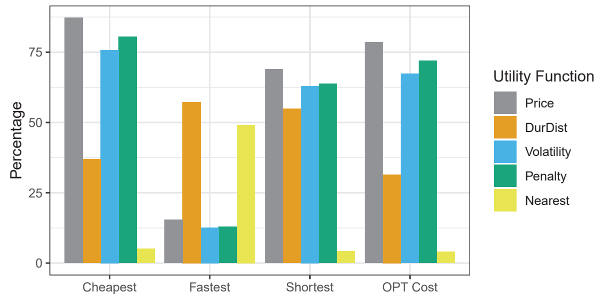

Fig. 4. Percentage for each utility function depicted in different colors of selecting the cheapest, fastest, shortest, and cost-optimal route.

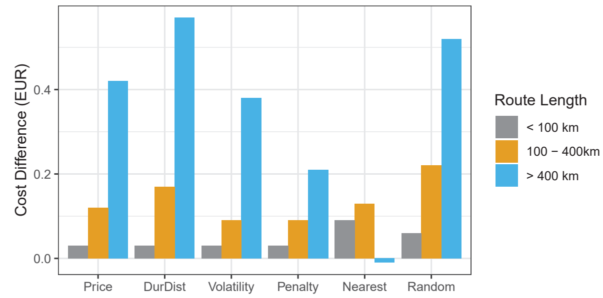

Fig. 5. Difference between estimated and actual costs for each utility function with regards to three categories of route lengths.

The duration-/distance utility function has a higher percentage deviation of around 1.6% (0.41 €). This is due to its characteristics of additionally integrating the duration to the distance of a planned route. The percentage deviation of the nearest station utility function is around 5% (1.13 €) and of the random utility function around 10% (2.25 €) per route. This can be explained by the nature of the utility functions, as they ignore any metric and choose the gas station randomly.

E. Planning with User Preferences

To analyze how user preferences might influence the adaptation of the planning logic, we use three different categories of preferences and measure how often each of the six applied utility functions selects the cheapest, fastest, or shortest route. Besides, we compare the solutions to the theoretical cost-optimum, i.e., how often each utility function selects the solution that would have been selected if all information was available previously (cf. Section IV-B).

Figure 4 shows the percentage for each utility function of selecting the cheapest, fastest, shortest, and cost-optimal route. The sum of all utility functions for one category does not necessarily sum up to 100% as multiple utility functions can select the same route and achieve the same score. When looking at the results for the cheapest, shortest, and cost-optimal percentages, the gray bar (price-aware utility function) always shows the highest percentage of around 70-85%, followed by the blue (volatility-aware utility function) and the green bar (penalty-aware utility function) around 65-80%. The orange bar (duration-/distance utility function) shows substantially lower percentages of around 35-55%, and the yellow (nearest station) and dark blue bars (random utility function) show very low percentages of around 6% and 0%, respectively. This means that the price-aware utility function performs best in terms of calculating the cheapest, shortest, and cost-optimal route. However, the duration-/distance and nearest station utility functions perform better w.r.t. the selection of the fastest route. This happens since this utility function is the only one considering the duration factor in its calculations and by selecting the nearest gas station, the incurred detour is very low. These results show that user preferences have an impact on the selection of the optimal utility function.

F. Planning under Uncertainty

Our hypothesis of increasing uncertainty with a longer planning horizon implies that the longer the planning horizon, i.e., the longer the route, the larger the difference between estimated and actual costs. Therefore, we categorized the routes into three categories: (i) shorter than 100 km, (ii) between 100 km and 400 km, and (iii) longer than 400 km.

Figure 5 visualizes the difference between estimated and actual costs for each utility function with regards to the three categories of route lengths. The gray bars ($<$ 100 km) show for all utility functions except the nearest station the smallest difference between estimated and actual costs of around 0.03 € and 0.09 €. The orange bars (100 - 400 km) show a noticeable increase in cost difference between 0.09 € and 0.22 €. The blue bars ($>$ 400 km) show especially for the duration-/distance and random utility function strong increases in cost differences between 0.52 € and 0.56 € while the cost increase for the other utility functions is less significant. The nearest station utility function does not reflect these characteristics as only a slight increase can be shown between short and medium route lengths, and is even reduced slightly for long routes. This can be explained by the selection criterion since the travel time to the gas station is short and price changes occur very rarely. These results indicate that the size of the planning horizon and that integrating uncertainty parameters into utility functions is necessary to handle unforeseen situations.

G. Threats to Validity

We identified the following threats to the validity of the evaluation results. First, since a diverse set of objectives lies in the nature of vehicle routing, we cannot cover the entire area. However, we focused on the most meaningful objectives and showed that even these intuitive utility functions enable us to assess the effects of uncertainty and differing user preferences. Second, the category opt-cost could bias the evaluation results as it is only based on the gas price and hence might favor the price-aware utility function. Nevertheless, we have chosen the opt-cost utility function for our comparison as we want to inspect the uncertainty in terms of costs per route. Third, we studied the user preferences of the cheapest, fastest, and shortest routes. However, these categories represent the most common user preferences provided by navigation systems. Fourth, our set of dates and routes used for the evaluation is limited to 18 dates within one year and 22 routes. However, we carefully selected dates regarding holidays or vacations, working days, and weekends and the 22 routes represent routes that cover Germany to a large extent. Finally, our tool CostSAVeR supports self-adaptivity at runtime, i.e., the adaptation of the route while driving in response to changing gas prices. As it is hard to evaluate this self-adaptation due to the dynamic environment (such as changing road traffic), we decided to omit the evaluation of adaptations at runtime.

V. Conclusion

In this paper, we studied the effects of adaptation planning under uncertainty induced by the non-deterministic environment or the requirement to adapt to the preferences of possibly changing user groups, i.e., integrating human-in-the-loop [2]. To asses our hypotheses, we used CostSAVeR, a self-adaptive route calculation system that applies multi-criteria optimization for cost-aware routing, for comparing utility functions from three categories: (i) two utility functions incorporate the gas price, distance, and duration, (ii) two utility functions try to cope with the uncertainty of volatile gas prices, and (iii) two utility functions serve as a comparison and select the nearest gas station or completely random ones. The uncertainty analysis showed that the time horizon for proactive planning impacts the quality of the result significantly, i.e., the wider the planning horizon the higher the deviation of the estimated to the actual costs. This supports our hypotheses that uncertainty as well as user preferences are essential aspects for adaptation planning.

Based on our findings and the results of our use case study, we identified the following future work. Using existing taxonomies on uncertainty (e.g., [2], [19], [20]), we plan to provide a model for identifying necessary characteristics of a situation w.r.t. uncertainty. A taxonomy will accompany this for describing how specific utility functions tackle uncertainty. This supports the mapping of uncertainty functions to uncertainty in a situation, which is the first step towards meta-adaptive planning under uncertainty. Finally, an interesting comparison could be to compare our approach to integrating predictions and forecasting (e.g., based on [21]) into the decision making process for taming uncertainty. This can also help to learn the reasons for uncertainty and improve the utility functions at runtime.

References

[1] M. Weiser, "The Computer for the 21st Century," *Scientific American*, vol. 265, no. 3, pp. 94–104, 1991.

[2] S. Mahdavi-Hezavehi, P. Avgeriou *et al.*, "A Classification of Current Architecture-based Approaches Tackling Uncertainty in Self-Adaptive Systems with Multiple Requirements," in *Managing Trade-offs in Adaptable Software Architectures*. Elsevier, 2016.

[3] C. Fitzgerald, B. Klöpper *et al.*, "Utility-Based Self-Adaption with Environment Specific Quality Models," in *Adaptive and Intelligent Systems*, A. Bouchachia, Ed. Springer, 2011, pp. 107–118.

[4] K. Kakousis, N. Paspallis *et al.*, "Optimizing the Utility Function-Based Self-adaptive Behavior of Context-Aware Systems Using User Feedback," in *On the Move to Meaningful Internet Systems: OTM 2008*, R. Meersman and Z. Tari, Eds. Springer, 2008, pp. 657–674.

[5] S. Shevtsov, D. Weyns *et al.*, "SimCA*: A Control-Theoretic Approach to Handle Uncertainty in Self-Adaptive Systems with Guarantees," *ACM TAAS*, vol. 13, no. 4, 2019.

[6] J. Cámara, D. Garlan *et al.*, "Evaluating Trade-Offs of Human Involvement in Self-Adaptive Systems," in *Managing Trade-Offs in Adaptable Software Architectures*. Morgan Kaufmann, 2017, pp. 155 – 180.

[7] M. Gil, V. Pelechano *et al.*, "Designing the Human in the Loop of Self-Adaptive Systems," in *Ubiquitous Computing and Ambient Intelligence*, C. R. García, P. Caballero-Gil *et al.*, Eds. Springer, 2016, pp. 437–449.

[8] S. Huang and P. Miranda, "Incorporating Human Intention into Self-Adaptive Systems," in *Proc. ICSE*, 2015, pp. 571–574.

[9] C. Becker, J. Hähner *et al.*, "Flexibility in Organic Systems - Remarks on Mechanisms for Adapting System Goals at Runtime," in *Proc. ICINCO*, 2012.

[10] J. Cámara, G. A. Moreno *et al.*, "Reasoning about human participation in self-adaptive systems," in *Proc. SEAMS*, 2015, pp. 146–156.

[11] I. Gerostathopoulos, C. Prehofer *et al.*, "Adapting a System with Noisy Outputs with Statistical Guarantees," in *Proc. SEAMS*, 2018.

[12] Y. Zhao, S. Yao *et al.*, "GreenRoute: A Generalizable Fuel-Saving Vehicular Navigation Service," in *Proc. ICAC*, 2019.

[13] S. Tomforde, H. Prothmann *et al.*, "Decentralised Progressive Signal Systems for Organic Traffic Control," in *Proc. SASO*, 2008.

[14] C. Krupitzer, J. Otto *et al.*, "Adding Self-Improvement to an Autonomic Traffic Management System," in *Proc. ICAC*, 2017, pp. 209–214.

[15] C. Krupitzer, F. M. Roth *et al.*, "A Survey on Engineering Approaches for Self-Adaptive Systems," *PMCJ*, vol. 17, no. Part B, 2015.

[16] J. O. Kephart and D. M. Chess, "The Vision of Autonomic Computing," *IEEE Computer*, vol. 36, no. 1, pp. 41–50, 2003.

[17] W. E. Walsh, G. Tesauro *et al.*, "Utility Functions in Autonomic Systems," in *Proc. ICAC*, 2004, pp. 70–77.

[18] A. Zheng and A. Casari, *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*. O'Reilly, 2018.

[19] A. J. Ramirez, A. C. Jensen *et al.*, "A Taxonomy of Uncertainty for Dynamically Adaptive Systems," in *Proc. SEAMS*, 2012, pp. 99–108.

[20] G. A. Moreno, J. Cámara *et al.*, "Uncertainty Reduction in Self-Adaptive Systems," in *Proc. SEAMS*, 2018, pp. 51–57.

[21] M. Züfle, A. Bauer *et al.*, "Autonomic Forecasting Method Selection: Examination and Ways Ahead," in *Proc. ICAC*, 2019.