Dieter Schuller

# QoS-aware Service Selection – An Overview of Optimization Mechanisms and Decision Support for Complex Service-based Workflows

**Abstract:** The problem of selecting services from a set of functionally appropriate ones based on non-functional service attributes such as cost or Quality of Service is well-recognized in the literature. It is commonly referred to as *Service Selection Problem*. Related approaches thereby only address process structures and workflow patterns that are rather straightforward. The approach presented in the work at hand accounts for this limitation, considering complex, interlaced, acyclic as well as cyclic structured and unstructured workflows that had not been considered to date. In addition, an approach accounting for stochastic Quality of Service behavior is described.

**Keywords:** Service Selection Problem, Stochastic Quality of Service, Optimization, Simulation

**Dieter Schuller:** E-Mail: dieter.schuller@KOM.tu-darmstadt.de

## I Introduction

In competitive markets with similar products and services, enterprises are facing various challenges, such as dynamically changing business environments and tough cost pressure. Due to changing customer and business requirements, a high degree of flexibility and efficiency with respect to developing and creating value-added products and services is indispensable. Thus, corresponding business processes need to be flexibly adaptable and efficient. The need for flexibility and efficiency is further intensified by the globalization and corresponding global, highly competitive markets.

In order to cope with these challenges, the application of a service-oriented approach that aims at achieving servicebased processes and workflows and therewith enables the creation of flexible processes is often recommended [1], [2]. Using *services* for supporting and realizing (service-based) business processes thereby constitutes the key concept [3]. From a business point of view, services represent and encapsulate business functionality, providing business logic [4]. From a technical point of view, services encapsulate and provide computation logic, constituting the IT-representation of business functionality [3]. Using services, business processes and workflows can be flexibly created and adapted according to changing customer and business requirements. The services thereby do not necessarily have to be located within or implemented by the own enterprise. Following the vision of the "Internet of Services", services addressing different areas of life and business will be available [5], [6]. They will be offered at service marketplaces by various service providers, so that enterprises can decide whether such services can be beneficially involved in and invoked for executing own business processes. If services providing equally appropriate functionalities are offered at such marketplaces by different service providers at different cost and quality levels, enterprises can select between them based on non-functional requirements, i.e., based on required cost and quality levels, for realizing steps and tasks within own business processes. Selecting services at minimal cost satisfying the enterprise's demands with respect to the considered quality attributes achieves realizing efficient processes.

The problem of selecting services from a set of functionally appropriate ones based on non-functional service attributes such as cost or Quality of Service (QoS) is well-recognized in the literature. It is commonly referred to as *Service Selection Problem* (SSP). The work at hand provides an overview of approaches and results provided in my Ph.D. thesis [7] in the area of service selection accounting for deterministic as well as for stochastic non-functional service attributes.

While many related approaches exist addressing the SSP and providing optimal as well as heuristic solutions to it, the process structures and workflow patterns considered and supported are rather straightforward. This limitation is addressed in [7] focusing on complex, interlaced, acyclic as well as cyclic structured and unstructured workflows, that so far have not been accounted for. It is thereby aimed at computing optimal solutions to the SSP considering *deterministic* non-functional service attributes. An overview of the proposed optimization approach accounting for deterministic attributes is provided in Section II.

Since QoS is not necessarily always deterministic, values for such non-functional service attributes may differ. Thus, even when the selected services, which have been obtained by applying the aforementioned optimal optimization approach, satisfy the specified QoS constraints, such constraints may still be violated at runtime, i.e., when the selected services are actually executed. Thus, because these services possibly show a different QoS behavior at runtime, corresponding constraints might still be violated. This potentially leads to severe consequences such as penalties. An overview of the developed adaptation strategies that focus on reducing the risk of uncertainty and therewith of a potential negative impact of stochastic QoS behavior is provided in Section III.

Finally, conclusions are drawn in Section IV.

# II Optimizing Qos for complex (un)-structured Workflows

In this section, the approach for computing optimal solutions to the SSP accounting for deterministic non-functional service attributes as well as the developed optimization framework enabling such computations are briefly introduced in Section II-A. A differentiation from related work is given in Section II-B. Finally, an overview of obtained results is provided in Section III-C.

## A Optimization Framework for QoS-aware Service Selection

Regarding the proposed solution approach for solving the SSP, the goal is to develop a means for enabling automated computations of *optimal* solutions to the SSP accounting for the mentioned complex workflows without having to enumerate all possible solutions to the problem for finding the optimal one, if such a solution exists. The proposed solution approach supports the development of heuristic solution methods to the extent that comparisons of the solution quality achieved by heuristic solution methods are enabled without the mentioned necessity for enumerating all possible solutions. To achieve this, an *Optimization Framework for QoS-aware Service Selection* has been developed.

For finding solutions to the SSP, it is firstly modeled as optimization problem. In this respect, *aggregation specifications* for aggregating values on non-functional service attributes of the candidate services have been developed. Applying these aggregation specifications according to the workflow structure that has to be considered for the optimization enables the creation of objectives and constraints – and therewith of a corresponding optimization model. However, depending on the considered QoS attributes, the created optimization model potentially constitutes a non-linear optimization model due to non-linear aggregations of *decision variables* that indicate whether a certain service candidate is selected or not. Thus, corresponding *linearization techniques* have been developed in order to enable transforming nonlinear optimization models that can result from applying aforementioned aggregation specifications into *linear* ones, if necessary. Once the obtained optimization model is linear, optimal solutions can be computed by applying standard *Linear Programming* (LP) techniques (cf., e.g., [8], [9]) provided by readily available LP solvers such as *CPLEX* [10] or *lpsolve* [11].

## B Differentiation from Related Work

In order to separate the approach proposed in [7] from related work, the workflow patterns considered and addressed by related approaches as well as the size of the optimization problem that has to be solved for finding an optimal solution to the SSP are used. A comparison of absolute computation time values in, e.g., milliseconds (msec), is omitted as such values strongly depend on concrete implementations of the corresponding approach as well as on the used hardware and programming environment.

In Table I, an overview of addressed workflow patterns is provided. The checkmarks thereby indicate whether the corresponding patterns are addressed by the related approach, i.e., the absence of a checkmark indicates that the corresponding pattern has not been addressed. If a certain pattern is only addressed to a limited extent, a bracketed checkmark is used. Textual descriptions explaining the corresponding limitations can be found in [7]. Further, it has to be noted that only the most commonly cited and referenced related work is listed in Table I.

As indicated in Table I, none of the related work is capable of addressing workflows that contain *OR-blocks*, unstructured *Directed Acyclic Graphs* (DAG) as well as unstructured *Single Entry Multiple Exit* (SEME) loops, which distinguishes the work in [7] from related work in this field. In addition, *XOR-blocks* as well as *Repeat loops* (RL) are only addressed to a limited extent by related approaches.

## C Results

The proposed optimization approach enables computing optimal solutions to the SSP without having to enumerate the complete solution space, as previously stated. For this, the developed optimization framework creates an optimization problem according to the workflow that is subject for optimization and transforms it into a linear one, if necessary.

Evaluation results show that the proposed approach for computing optimal solutions to the SSP, which is actually based on *Integer Linear Programming* (ILP) that is known to be NP-complete in the worst case, is not "much worse" in terms of computation time compared to using *Mixed Integer Linear Programming* (MILP), which is known to compute solutions in polynomial time. For instance, regarding a workflow with 10 tasks and 20 candidate services per task, the proposed ILP-based optimization approach takes 29.4 msec, whereas using MILP leads to a computation time of 27.2 msec.

In this respect, it has to be noted that it cannot be guaranteed to obtain valid solutions to the SSP when using MILP, since a MILP-based optimization approach does not make sure that exactly one service is selected for each task of the workflow that has to be optimized. Thus, from a "functional" perspective, the computation time of the proposed ILPbased optimization approach may not be compared with the computation time achieved by MILP-based approaches. In fact, the computation time should be compared with other approaches capable of ensuring that the computed (optimal) solution to the SSP constitutes a *valid* one.

As indicated in Table I, my proposed optimization approach [7] is capable of addressing workflows that contain OR-blocks, unstructured DAGs as well as unstructured SEME loops – which are not supported by related approaches – in addition to sequences, AND-/XOR-blocks and Repeat loops. Thus, due to the lack of available related approaches, the proposed optimization approach should actually be compared to an exhaustive enumeration approach, which naively enumerates and checks all possible solutions for finding an optimal one, because this approach is to date the only related approach capable of computing valid, optimal solutions to optimization problems accounting for aforementioned patterns. Evaluation results show that even for small problem instances, the exhaustive enumeration approach takes a disproportionately high computation time, while the proposed ILP-based approach only requires a few milliseconds. For instance, computing an optimal solution to the aforementioned workflow with 10 tasks and 20 candidate services per task takes 29.4 msec when using the proposed ILP-based approach, whereas the exhaustive enumeration approach would require approximately three years.

# III Adaptation Strategies For Addressing Stochastic Qos Behavior

While the optimization approach presented in the previous section considers deterministic non-functional service attributes, this section focuses on stochastic QoS behavior.

**Table I:** Addressed workflow patterns of optimal solution approaches

| Approach | Seq | AND | XOR | RL | OR | DAG | SEME |
|---|---|---|---|---|---|---|---|
| Anselmi et al. [12] | ✓ | ✓ | (✓) | (✓) | | | |
| Ardagna and Pernici [13]–[15] | ✓ | ✓ | (✓) | (✓) | | | |
| Cardellini et al. [16]–[18] | ✓ | ✓ | (✓) | | | | |
| Gao et al. [19]–[21] | ✓ | ✓ | (✓) | (✓) | | | |
| Huang et al. [22] | (✓) | (✓) | (✓) | (✓) | | | |
| Jiang et al. [23] | (✓) | (✓) | (✓) | (✓) | | | |
| Klein et al. [24] | ✓ | ✓ | (✓) | (✓) | | | |
| Menasce et al. [25], [26] | ✓ | ✓ | ✓ | ✓ | | | |
| Meng et al. [27] | ✓ | ✓ | | | | | |
| Yu et al. [28] | ✓ | ✓ | (✓) | (✓) | | | |
| Zeng et al. [29] | ✓ | ✓ | (✓) | (✓) | | | |
| Schuller [7] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

An overview of the stochastic QoS model accounting approach proposed in [7] is provided in Section III-A. In Section III-B, a differentiation from related work will be conducted, while an overview of obtained results is presented in Section III-C.

## A Stochastic QoS Model Accounting Approach

In their work in [30], Rosario et al. provide measurement records for a certain Web service that show that the response time of this Web service is differing over time. Also the authors in [31]–[33] provide evidence that QoS differs over time. Miede et al., for instance, empirically showed substantial fluctuations in *response time* for different Web services, while Lampe et al. [33] additionally accounted for availability and throughput of different services. Such differing QoS values can be traced back to network latency or server load, i.e., to different data transferring times or different request processing times at the server side [34].

Thus, since QoS values might change at runtime, which has been shown by various studies (cf., e.g., [30]–[34]), the perceived QoS might differ from the QoS expected at design time when the SSP has been modeled as optimization problem and an optimal solution to it has been computed. Such potential differences may lead to unexpected – and at worst – to undesired situations, in which constraints on QoS will still be violated, although an optimal solution to the SSP has been computed beforehand that satisfied the constraints. Depending on the impact of such QoS violations, severe consequences may result. For instance, accounting for a scenario where a service broker receives requests from his/her customers who require certain tasks and workflows, respectively, to be executed, the broker takes responsibility for selecting services, invoking them, and therewith executing the specified workflows, for which the customers pay the broker a fixed amount of money. Of course, the customers have requirements on non-functional properties in addition to functional ones. Thus, they demand certain QoS levels to be satisfied with respect to executing their requested workflows. Violating the customers' QoS requirements is assumed to lead to penalties, i.e., the broker will be penalized and will have to pay certain penalty fees. In order to cope with such negative consequences, a simulation-based adaptation framework along with adaptation strategies and corresponding adaptation heuristics has been developed in [7]. It explicitly accounts for dynamically changing and therewith for stochastic QoS behavior, aiming at reducing negative impacts of differing QoS such as penalties. The framework realizes an integrated three-step approach, comprising an optimization, a simulation, and an adaptation step. While optimal solutions to the SSP are computed in the first step by applying the optimization approach proposed in [7], these solutions are simulated in the second step in order to determine QoS behavior and to assess its impact on constraint violation. If undesired QoS behavior is detected, an adaptation step will be carried out which aims at reducing the impact of the determined QoS behavior.

In this respect, different adaptation strategies and corresponding heuristics, implementing these strategies, are proposed and provided in [7]. In effect, the strategies basically try to exchange services with high uncertainties for services with rather low uncertainties by banning corresponding services from the list of possible candidate services for the different workflow steps depending on their impact on QoS constraint violation. Using the adapted list of candidate services, another optimization step is carried out, leading to a new “optimal” solution – according to the adapted list of candidate services and therewith to the pruned list of available services. Subsequently, another simulation step is conducted in order to determine QoS behavior again and to assess its impact on constraint violation. If it is found out that another adaptation step is necessary, the corresponding adaptation heuristics will be applied again, so that another iteration of the described three-step approach will be carried out. This procedure is repeated until a satisfying compromise between service invocation and penalty cost is achieved.

Evaluation results show that reductions in total cost (comprising invocation and penalty cost) up to 30% can be achieved by reducing penalty costs that accrue due to the violation of QoS constraints – depending on the considered scenario.

## B Differentiation from Related Work

Related work considering stochastic, non-functional service attributes ranges from determining and applying mean values by considering quantile values up to applying and aggregating statistical distribution functions. The work in [7] thereby primarily focuses on reducing potential negative consequences, such as additionally accruing penalty cost due to violating QoS constraints. Thus, for differentiating the work in [7] from related work, it will be mainly focused on whether and how (potentially negative) effects of stochastic QoS is considered. In contrast to related approaches that consider deterministic non-functional service attributes for computing solutions to the

SSP, related work accounting for stochastic attributes is rather sparse.

In summary, related approaches considering stochastic non-functional service attributes do not sufficiently account for (potentially negative) consequences caused by stochastic QoS behavior. While several authors focus on aggregating distribution functions (cf., e.g., [34], [35]) and, thus, do not address the SSP, others consider stochastic nonfunctional service attributes to the extent that they predict values for them. These predicted, expected values, which actually constitute fixed values, are then used for the optimization. The authors in [36] address stochastic QoS by considering percentile-based constraints. But they do not pay attention to negative consequences resulting from violating QoS constraints. The work in [37] addresses potentially arising consequences resulting from stochastic QoS, but the authors only consider stochastic QoS behavior for single services in isolation, whereas the work in [7] accounts for potential reverse QoS deviations of different services that probably compensate each other. This way, the impact of stochastic QoS behavior is considered with respect to the whole workflow.

Thus, by accounting for potential negative consequences resulting from stochastic QoS behavior as well as by considering potential reverse QoS deviations balancing each other, the work in [7] extends related work in this field.

### C Results

For evaluating the proposed stochastic QoS model accounting approach and assessing its capability of reducing potential negative effects of stochastic QoS behavior, different scenarios have been considered. In this context, corresponding optimization problems have been generated and solved optimally, i.e., respective services have been selected, without applying any adaptation approach. The resulting solution quality, which is represented in this case by the accruing total cost, has then been compared to the corresponding solution quality obtained by applying the developed stochastic QoS model accounting approach. Applying the corresponding adaptation heuristics thereby successfully achieved reducing the negative impact of stochastic QoS behavior.

Using the developed simulation-based adaptation framework, significant reductions in total cost, which comprise service invocation cost as well as penalty cost accruing due to QoS violations, have already been achieved after only 2–4 iterations of the described three-step approach. For instance, accounting for a workflow with eight workflow steps containing an interlaced AND-block, total cost has been reduced from 430.2 monetary units, which accrue when applying the “optimal” solution to the SSP without further adaptation, to 380.1 monetary units after two iterations of the proposed stochastic QoS model accounting approach. This corresponds to a reduction by 11.6%. After six iterations, a reduction by 15.3%, i.e., to 364.3 monetary units, has been realized. For other scenarios and workflows with, e.g., 14 process steps, containing SEME loops and interlaced OR-blocks, reductions up to approximately 30% have been achieved.

In this respect, it has to be noted that the presented absolute as well as relative cost reductions strongly depend on the considered scenario. However, applying the proposed stochastic QoS model accounting approach successfully reached the goal of reducing potential negative consequences that arose due to stochastic QoS behavior.

## IV Conclusion

In competitive markets with similar products and services, business processes need to be flexible and efficient. In this respect, my Ph.D. thesis [7] proposed an approach that aims at optimizing service-based workflows with respect to non-functional service properties, i.e., with respect to QoS and cost, by selecting services among functionally appropriate ones for accomplishing the tasks of such servicebased workflows. The problem of selecting services among functionally appropriate ones based on non-functional service properties is actually well-recognized in the literature. It is commonly referred to as (QoS-aware) *Service Selection Problem*, respectively. For addressing and solving the SSP, an optimization framework has been developed (cf. Section II-A) that enables specifying the SSP as a (linear) optimization problem according to the workflow that is subject for optimization. The work in [7] thereby focuses on complex, interlaced, acyclic as well as cyclic structured and unstructured workflows, that so far have not been considered by related work. The corresponding solution approach for finding optimal solutions to the SSP by applying *Integer Linear Programming* constitutes the main contribution of the work in [7]. In addition to the proposed ILP-based solution approach, a heuristic solution approach has been developed for addressing scalability issues.

Evaluation results show that the proposed ILP-based approach for computing optimal solutions to the SSP is not performing “much worse” regarding required computation time compared to an approach applying MILP. However, MILPbased approaches cannot ensure that the provided solution is valid. Thus, due to unavailability of related approaches capable of ensuring that the found (optimal) solution is valid, the proposed ILP-based approach should actually be compared to an exhaustive enumeration approach, as described in Section II-C, which requires unacceptable high computation times for larger problem sizes.

As shown in various studies in the context of Web services (cf., e.g., [30]–[34]), QoS is not necessarily always deterministic. Thus, constraints on non-functional service attributes may still be violated at runtime and lead to severe consequences such as penalties even if an optimal solution has been computed beforehand for which the constraints were satisfied. In order to assess and reduce potentially negative consequences of differing QoS, a simulation-based adaptation framework along with adaptation strategies and corresponding adaptation heuristics has been developed in my Ph.D. thesis [7].

Evaluation results show that a significant decrease in total cost can be achieved when applying the proposed adaptation heuristics – up to 30% in the considered scenarios. While the absolute as well as the relative values indicating the cost reduction strongly depend on the considered scenario, the conducted evaluation nevertheless provides evidence that a potential negative impact of stochastic QoS behavior can be reduced by applying the proposed stochastic QoS model accounting approach, thus meeting the initial objective.

In summary, the two main contributions provided in [7] are the described approach for computing optimal solutions to the SSP considering various complex, structured as well as unstructured workflows, and the approach for addressing stochastic QoS in the context of the SSP aiming at reducing negative consequences of unexpected and undesired QoS behavior. Thus, a means for optimal service selection based on *deterministic* non-functional service attributes as well as an approach for reducing negative impacts of *stochastic* nonfunctional service attributes has been provided.

## Acknowledgement

This work is supported in part by the Commission of the European Union within the ADVENTURE FP7-ICT project (Grant agreement no. 285220), and by E-Finance Lab e.V., Frankfurt am Main, Germany (http://www.efinancelab.com).

## References

1 M. P. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann, "Service-oriented Computing," *Communications of the ACM*, vol. 46, no. 10, pp. 25–28, 2003.
2 D. Georgakopoulos and M. P. Papazoglou, *Service Oriented Computing*, ser. Cooperative Information Systems Series. Cambridge, MA, USA: MIT Press, 2009.
3 N. Josuttis, *SOA in Practice: The Art of Distributed System Design*. Cambridge, MA, USA: O'Reilly Media, 2007.
4 T. Erl, Service-Oriented Architecture: Concepts, Technology, and Design. Upper Saddle River, NJ, USA: Prentice Hall, 2005.
5 C. Schroth, "The Internet of Services: Global Industrialization of Information intensive Services," in International Conference on Digital Information Management (ICDIM 2007), 2007, pp. 635–642.
6 D. Oberle, N. Bhatti, S. Brockmans, M. Niemann, and C. Janiesch, "Countering Service Information Challenges in the Internet of Services," Business & Information Systems Engineering, vol. 1, no. 5, pp. 370–390, 2009.
7 D. Schuller, QoS-aware Service Selection – Optimization Mechanisms and Decision Support for Complex Servicebased Workflows. Munich, Germany: Verlag Dr. Hut, 2013.
8 B. Aspvall and R. E. Stone, "Khachiyan's Linear Programming Algorithm," Journal of Algorithms, vol. 1, no. 1, pp. 1–13, 1980.
9 G. B. Dantzig, Linear Programming and Extensions, ser. Princeton Landmarks in Mathematics and Physics: Mathematics. Princeton, NJ, USA: Princeton University Press, 1998.
10 IBM, "Ilog CPLEX optimization studio," last access on 2013-05-06. [Online]. Available: http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/index.html.
11 M. Berkelaar, J. Dirks, K. Eikland, P. Notebaert, and J. Ebert, "Lp solve," last access on 2013-05-06. [Online]. Available: http://lpsolve.sourceforge.net/5.5/.
12 J. Anselmi, D. Ardagna, and P. Cremonesi, "A QoS-based Selection Approach of Autonomic Grid Services," in Workshop on Service-Oriented Computing Performance (SOCP 2007), 2007, pp. 1–8.
13 D. Ardagna and B. Pernici, "Global and Local QoS Guarantee in Web Service Selection," in Workshop on Business Process Management (BPS 2005), 2005, pp. 32–46.
14 Danilo Ardagna and Barbara Pernici, "Global and local QoS constraints guarantee in Web service selection," in International Conference on Web Services (ICWS 2005), 2005, pp. 805–806.
15 D. Ardagna and B. Pernici, "Adaptive Service Composition in Flexible Processes," IEEE Transactions on Software Engineering, vol. 33, no. 6, pp. 369–384, 2007.
16 V. Cardellini, E. Casalicchio, V. Grassi, and F. L. Presti, "Flow-Based Service Selection for Web Service Composition Supporting Multiple QoS Classes," in International Conference on Web Services (ICWS 2007), 2007, pp. 743–750.
17 V. Cardellini, V. D. Valerio, V. Grassi, S. Iannucci, and F. L. Presti, "A New Approach to QoS Driven Service Selection in Service Oriented Architectures," in International Symposium on Service Oriented System Engineering (SOSE 2011), 2011, pp. 102–113.
18 V. Cardellini, V. Di Valerio, V. Grassi, S. Iannucci, and F. L. Presti, "A Performance Comparison of QoS-Driven Service Selection Approaches," in European Conference Towards a Service-based internet (ServiceWave 2011), 2011, pp. 167– 178.

19 Y. Gao, B. Zhang, J. Na, L. Yang, Y. Dai, and Q. Gong, "Optimal Selection of Web Services for Composition Based on Interface-Matching and Weighted Multistage Graph," in International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT 2005), 2005, pp. 336–338.

20 Y. Gao, B. Zhang, J. Na, L. Yang, and Y. Dai, "Optimal Selection of Web Services for Composition Using Weighted Multistage Graph," in International Conference on Information Integrationand Web-based Applications Services (iiWAS 2005), 2005, pp. 25–34.

21 Y. Gao, J. Na, B. Zhang, L. Yang, and Q. Gong, "Optimal Web Services Selection Using Dynamic Programming," in International Symposium on Computers and Communications (ISCC 2006), 2006, pp. 365–370.

22 A. F. M. Huang, C.-W. Lan, and S. J. H. Yang, "An Optimal QoS-based Web Service Selection Scheme," Information Sciences, vol. 179, no. 19, pp. 3309–3322, 2009.

23 W. Jiang, C. Zhang, Z. Huang, M. Chen, S. Hu, and Z. Liu, "QSynth: A Tool for QoS-aware Automatic Service Composition," in International Conference on Web Services (ICWS 2010), 2010, pp. 42–49.

24 A. Klein, F. Ishikawa, and S. Honiden, "Efficient QoS-Aware Service Composition with a Probabilistic Service Selection Policy," in International Conference on Service Oriented Computing (ICSOC 2010), 2010, pp. 182–196.

25 D. A. Menasce, E. Casalicchio, and V. K. Dubey, "A Heuristic' Approach to Optimal Service Selection in Service Oriented Architectures," in International Workshop on Software and Performance (WOSP 2008), 2008, pp. 13–24.

26 D. A. Menasce, E. Casalicchio, and V. Dubey, "On optimal' Service Selection in Service Oriented Architectures," Performance Evaluation, vol. 67, no. 8, pp. 659–675, 2010.

27 S. Meng and F. Arbab, "QoS-driven Service Selection and Composition," in International Conference on Application of Concurrency to System Design (ACSD 2008, 2008, pp. 160– 169.

28 T. Yu, Y. Zhang, and K.-J. Lin, "Efficient algorithms for Web services selection with end-to-end QoS constraints," ACM Transactions on the Web, vol. 1, no. 1, pp. 1–26, 2007.

29 L. Zeng, B. Benatallah, A. H. H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, "QoS-Aware Middleware for Web Services Composition," IEEE Transactions on Software Engineering, vol. 30, no. 5, pp. 311–327, 2004.

30 S. Rosario, A. Benveniste, S. Haar, and C. Jard, "Probabilistic QoS and Soft Contracts for Transaction-Based Web Services Orchestrations," IEEE Transactions on Services Computing, vol. 1, no. 4, pp. 187–200, 2008.

31 Z. Zheng, Y. Zhang, and M. R. Lyu, "Distributed QoS Evaluation for Real-World Web Services," in International Conference on Web Services (ICWS 2010), 2010, pp. 83–90.

32 A. Miede, U. Lampe, D. Schuller, J. Eckert, and R. Steinmetz, "Evaluating the QoS Impact of Web Service Anonymity," in European Conference on Web Services (ECOWS 2010), 2010, pp. 75–82.

33 U. Lampe, A. Miede, T. Lusa, S. Schulte, R. Steinmetz, and S. Dustdar, "An Analysis of Anonymity Side Effects in the Internet of Services," in International Conference on Networked Systems (NetSys 2013), 2013, pp. 51–58.

34 H. Zheng, J. Yang, and W. Zhao, "QoS Probability Distribution Estimation for Web Services and Service Compositions," in International Conference on Service-Oriented Computing and Applications (SOCA 2010), 2010, pp. 1–8.

35 S.-Y. Hwang, H. Wang, J. Tang, and J. Srivastava, "A Probabilistic Approach to Modeling and Estimating the QoS of Web Services based Workflows," Information Sciences, vol. 177, no. 23, pp. 5484–5503, 2007.

36 V. Cardellini, E. Casalicchio, V. Grassi, and F. L. Presti, "Adaptive Management of Composite Services under Percentile-Based Service Level Agreements," in International Conference on Service Oriented Computing (ICSOC 2010), 2010, pp. 381–395.

37 P. Leitner, W. Hummer, and S. Dustdar, "Cost-Based Optimization of Service Compositions," IEEE Transactions on Services Computing, no. PrePrints, pp. 1–14, 2011.


**Dieter Schuller:** Technische Universität Darmstadt – Multimedia Communications Lab (KOM), Rundeturmstr. 10, 64283 Darmstadt, Germany