Conventional and Machine Learning Approaches for Network and Service Coordination

Stefan Schneider
Paderborn University, Germany
stefan.schneider@upb.de

***Abstract*—Network and service coordination is important to provide modern services consisting of multiple interconnected components, e.g., in 5G, network function virtualization (NFV), or cloud and edge computing. In this paper, I outline my dissertation research, which proposes six approaches to automate such network and service coordination. All approaches dynamically react to the current demand and optimize coordination for high service quality and low costs. The approaches range from centralized to distributed methods and from conventional heuristic algorithms and mixed-integer linear programs to machine learning approaches using supervised and reinforcement learning. I briefly discuss their main ideas and advantages over other state-of-the-art approaches and compare strengths and weaknesses.**

I. Introduction

In various practical scenarios, services consist of multiple chained components, where each component is implemented in software and provides its own functionality. For example, components could be virtual network functions (VNFs) in network function virtualization (NFV) [1] and 5G [2], microservices in a service mesh for cloud and edge computing [3], or machine learning functions in a pipeline [4]. To deploy these services according to the current demand, each component can be instantiated multiple times across different nodes in the network. Nodes are distributed at different locations, where each node may represent a large data center, a small edge server, or a node without compute capacity. These nodes are interconnected by links with varying data rate limitations.

As user demand for the different services arrives at the network's ingress nodes, *network and service coordination* ensures that the requested services are deployed and provided with high quality and low costs. Particularly, I distinguish four main coordination aspects as illustrated in Fig. 1: 1) Service scaling, which determines the number of required instances per component. 2) Service placement, where suitable nodes are selected for deploying these instances. 3) Flow scheduling, where incoming flows are assigned to the placed instances. 4) Routing, which determines the flows' path between users and the assigned instances.

These four coordination aspects are inter-dependent and should therefore be optimized jointly [5]. As they depend on a multitude of parameters, it is challenging to ensure that enough resources are allocated for good service quality but also no resources and costs are wasted. With demand changing over time, coordination has to happen dynamically online.

To address these challenges, I propose approaches for automated and dynamic network and service coordination (Sec. II).

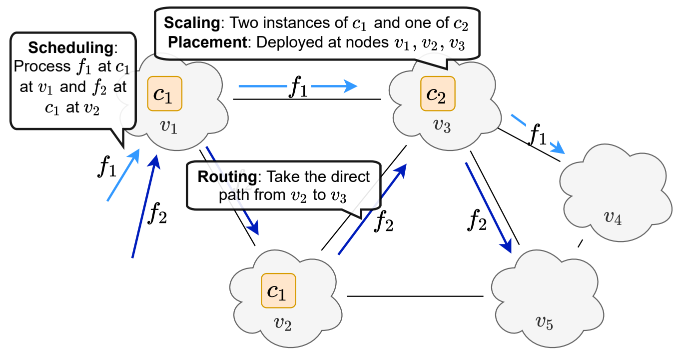

Fig. 1: Four main aspects of network and service coordination.

I group these approaches into either conventional approaches like heuristics and mixed-integer linear programs (MILPs) or machine learning approaches. In each group, I present both centralized and distributed approaches and discuss their strengths and weaknesses. Sec. III outlines currently ongoing and future work and concludes this paper.

II. Coordination Approaches

I provide a short overview of six coordination approaches, grouped into three conventional (Sec. II-A) and three machine learning approaches (Sec. II-B). I also briefly describe evaluation outcomes and reference corresponding publications for further details and information about involved co-authors.

A. Conventional Approaches

Conventional coordination approaches, e.g., heuristics or MILPs, are designed by experts and build on detailed models tailored to specific scenarios. They are suitable for scenarios that are well understood and can be modeled and observed accurately. Most existing work falls into this category.

1) Centralized Coordination: Existing approaches mostly address the four aforementioned coordination aspects separately, which easily leads to suboptimal results. Hence, I propose a centralized MILP and a heuristic approach to jointly optimize all four coordination aspects [6]. The approach also supports complex bidirectional service structures. Given enough time, the MILP can be used to for optimal coordination in small problem scenarios. The heuristic leverages its global view to find close-to-optimal solutions across a variety of scenarios but is much faster than the MILP.

2) Hierarchical Coordination: In large networks, detailed up-to-date global knowledge may not be available and centralized decisions may be too inefficient, even with a heuristic. Therefore, I propose a hierarchical coordination approach, which works in two phases [7]: In the bottom-up phase, it aggregates information from lower hierarchical levels, hiding unnecessary details from higher-level coordinators to reduce complexity. In the following top-down phase, higher-level coordinators make coarse-grained coordination decisions that are then refined at lower levels. Compared to an equivalent centralized approach, this hierarchical approach finds solutions with similar quality but is significantly faster.

3) Distributed Coordination: Going one step further, I propose two simple and fast heuristics for fully distributed coordination [8]. The algorithms are executed at each node in the network and rely only on local observations and control at each node. This allows fast-paced online coordination in practical large-scale networks with rapidly arriving flows. According to the evaluation, the fully distributed heuristics achieve similar solution quality but are magnitudes faster than a centralized approach.

B. Machine Learning Approaches

Machine learning coordination approaches do not require expert knowledge but learn coordination from available data or their own experience without human intervention.

1) Machine Learning for Dynamic Resource Allocation: Most conventional coordination approaches rely on given models (e.g., linear functions) to estimate the resource requirements of service components and to allocate resources relative to the current load. As it is hard to accurately model real-world components and their resource requirements manually, I propose an approach using supervised learning that automatically derives these models from available real-world benchmarking data [9]. I show that integrating the trained machine learning models into conventional algorithms requires low overhead but can significantly improve coordination performance.

2) Self-Learning Centralized Coordination: As an alternative to conventional approaches, I propose a completely autonomous self-learning coordination approach using model-free deep reinforcement learning (DRL) [10]. The approach trains a central DRL agent based on realistically available partial and delayed observations from monitoring. The DRL agent periodically updates coordination rules that are deployed at all network nodes and applied to incoming flows locally at runtime. In doing so, it outperforms existing conventional approaches but does not require detailed expert knowledge. It self-adapts to varying scenarios without human intervention, is robust to change, and scales to large networks.

3) Self-Learning Distributed Coordination: Finally, I propose a distributed self-learning coordination approach [11]. After centralized training, it makes fully distributed decisions locally at each node based on local observations, like the heuristics in Sec. II-A3, but using self-learning DRL agents rather than hand-written algorithms. Similar to the centralized DRL approach in Sec. II-B2, this approach self-adapts to varying scenarios and outperforms existing conventional approaches. Still, its distributed, scalable architecture allows much faster decisions and fine-grained control of individual flows, leading to overall better solutions.

III. Conclusion and Future Work

I presented six approaches to automate and optimize network and service coordination. The proposed conventional approaches should be used in well understood scenarios with clear underlying models. The machine learning approaches are useful in complex or constantly changing scenarios, where rigid models based on expert knowledge are unavailable or unsuitable. Centralized approaches can leverage their global view to achieve high solution quality. In large-scale scenarios without such global view, the proposed hierarchical or distributed approaches should be used. Overall, all proposed approaches effectively maximize service quality and minimize resource consumption and costs. Complementing this coordination in wired networks, I currently investigate challenges and approaches for coordination in wireless mobile networks [12].

Acknowledgments

This work was supported in part by the German Research Foundation within the Collaborative Research Centre “On-The-Fly Computing” (SFB 901) and the German Federal Ministry of Education and Research through Software Campus grant 01IS17046 (RealVNF).

References

[1] J. G. Herrera and J. F. Botero, “Resource allocation in NFV: A comprehensive survey,” *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 518–532, 2016.

[2] F. Z. Yousaf, M. Bredel, S. Schaller, and F. Schneider, “NFV and SDN—key technology enablers for 5G networks,” *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, 2017.

[3] C.-H. Hong and B. Varghese, “Resource management in fog/edge computing: A survey on architectures, infrastructure, and algorithms,” *ACM Computing Survey*, vol. 52, no. 5, Sep. 2019.

[4] ITU-T, “Architectural framework for machine learning in future networks including IMT-2020 (Y.3172),” 2019.

[5] L. Popa, G. Kumar, M. Chowdhury, A. Krishnamurthy, S. Ratnasamy, and I. Stoica, “FairCloud: Sharing the network in cloud computing,” in *ACM SIGCOMM Conference*, 2012, pp. 187–198.

[6] S. Dräxler, S. Schneider, and H. Karl, “Scaling and placing bidirectional services with stateful virtual and physical network functions,” in *IEEE Conference on Network Softwarization (NetSoft)*, 2018, pp. 123–131.

[7] S. Schneider, M. Jürgens, and H. Karl, “Divide and conquer: Hierarchical network and service coordination,” in *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2021.

[8] S. Schneider, L. D. Klenner, and H. Karl, “Every node for itself: Fully distributed service coordination,” in *International Conference on Network and Service Management (CNSM)*. IFIP/IEEE, 2020.

[9] S. Schneider, N. P. Satheeschandran, M. Peuster, and H. Karl, “Machine learning for dynamic resource allocation in network function virtualization,” in *IEEE Conference on Network Softwarization (NetSoft)*, 2020.

[10] S. Schneider, A. Manzoor, H. Qarawlus, R. Schellenberg, H. Karl, R. Khalili, and A. Hecker, “Self-driving network and service coordination using deep reinforcement learning,” in *International Conference on Network and Service Management (CNSM)*. IFIP/IEEE, 2020.

[11] S. Schneider, H. Qarawlus, and H. Karl, “Distributed online service coordination using deep reinforcement learning,” in *IEEE International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2021.

[12] S. Schneider, H. Karl, R. Khalili, and A. Hecker, “DeepCoMP: Coordinated multipoint using multi-agent deep reinforcement learning,” *https://ris.uni-paderborn.de/download/33854/33855/preprint.pdf*, 2022.