Intelligent Reconfiguration of Distributed Control of 6G Network Services

H. Shakespear-Miles, S. Barzegar, M. Ruiz, and L. Velasco*
Universitat Politècnica de Catalunya, Spain
e-mail: luis.velasco@upc.edu

Abstract: *Meeting network services performance requirements across multiple segments is especially challenging in dynamic mobility scenarios. We propose a distributed control system guaranteeing network services performance that reconfigures resources using mobility prediction, knowledge sharing and pre-training.*
Keywords: *Hybrid wireless-optical access networks including fixed mobile convergence and Metro access convergence*

I. INTRODUCTION

Guaranteeing the performance of Network Services (NS), e.g., in terms of end to end (e2e) delay, is not a straightforward task. Note that even though optical transport introduces low and quasi-deterministic latency, that from the packet network depends on the aggregated load of packet traffic leaving the output interfaces. That load heavily varies, especially in highly dynamic network scenarios [1]. For this very reason, a solution for ensuring the e2e performance of services with stringent requirements is to implement a near-Real-Time (nRT) control of the NS [2]. A possible solution is to rely on an intelligent distributed control architecture that relieves the Software-Defined Networking (SDN) control from nRT operation [3]. Such distributed control can be based on Multi-Agent Systems (MAS) [4], where a number of agents deployed close to the network devices make autonomous decisions for service assurance [5]. For instance, a solution for autonomous flow routing based on Deep Reinforcement Learning (DRL) was proposed in [6], where DRL *models* were trained offline and then fine-tuned during operation. The agents were coordinated by a centralized Machine Learning Function Orchestrator (MLFO) acting as a delegate of the Service Orchestrator (SO) for the nRT control of NSs, thus creating a MAS *pipeline*.

In this paper, we concentrate on mobility scenarios in e2e NSs supported by Radio Access (RAN) and packet-optical transport network segments. Here, it is important to anticipate NS reconfiguration to guarantee the committed performance. To that end, a special procedure is proposed to ensure that the MAS pipeline is ready to operate in short time. The procedure includes DRL-based traffic prediction models for the NS, which are trained by agents in the MAS. Such prediction model is shared and used by other agents to probe the resources allocated during the NS reconfiguration. Obtained results help to select the right pretrained model to be used for operation after the NS reconfiguration.

II. NETWORK SERVICE RECONFIGURATION

Fig. 1a illustrates the targeted mobility scenario, where the distributed MAS-based NS control reconfigures the NS and its related MAS pipeline in advance for service assurance. The top part of Fig. 1a reproduces the resources allocated to the NS connecting a drone, currently being served through RAN A1, to an application running in an edge/metro data center (DC) facility. A MAS pipeline oversees the service and guarantees that the delay in the transport network does not exceed some maximum value (denoted d_{max}).

In this paper, we concentrate on the actions made to guarantee d_{max}. Specifically, an agent (Agent 1) collocated with ingress packet switch S1 applies routing policies to balance routing among a set of routes that have been previously allocated by an SDN controller (in the example, routes P1 and P2 are available). The resulting delay is measured by another agent (Agent 2) collocated with egress packet switch (S3) that connects with the edge/metro DC. The MAS-based control can manage the delay component introduced by the transport network by balancing the traffic sent through the available routes [6]. Let us assume that the drone follows a trajectory that separates it from A1 and makes it closer to A2. If handover is performed at that time, only best effort connectivity will be provided, which could impact service performance. To avoid that, resources need to be allocated before handover happens. The bottom part of Fig. 1a shows that two new routes (P3 and P4) are allocated and a new agent (Agent 3) is deployed to control routing between packet switch S3 and the edge/metro DC.

III. PROPOSED NS RECONFIGURATION WORKFLOW

To anticipate handover with enough time so resources can be allocated (e.g., 10s [7]), an xApp running in the nRT RAN Intelligent Controller (RIC) follows the trajectory of the drone and triggers the reconfiguration of the NS, including the related MAS pipeline. Fig. 1b presents a simplified version of the proposed workflow. The xApp notifies the possible handover to the collocated Agent 1 in the MAS controlling the service (step 1 in Fig. 1b), which in turn

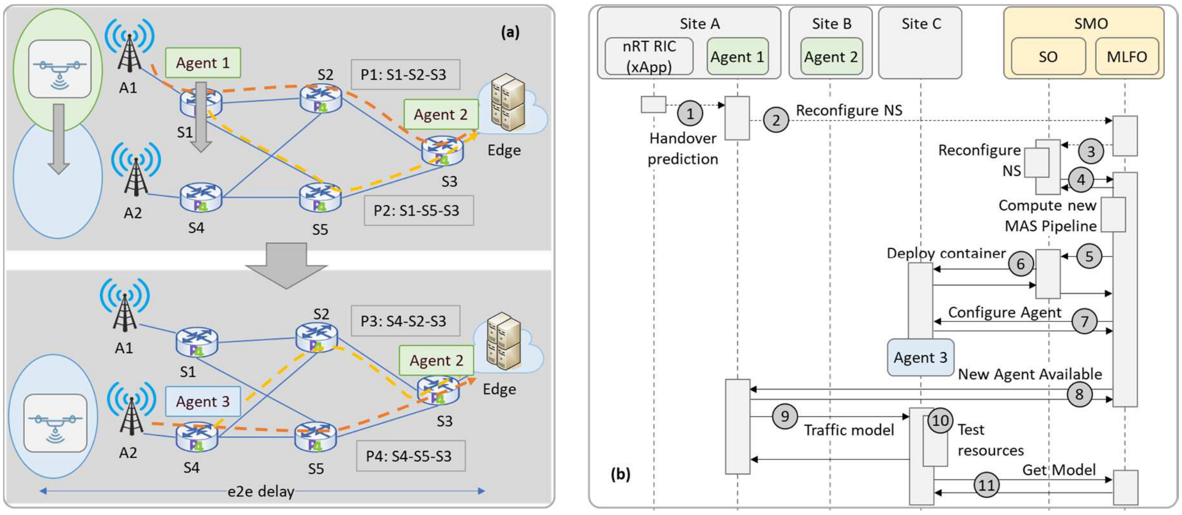

Fig. 1: Illustrative mobility scenario (a) and proposed workflow (b).

notifies the SO via the MLFO (2, 3). The SO finds the most appropriate forwarding graph for the NS including the predicted mobility and reconfigures the NS. In the example in Fig. 1a, the SO requests the new connectivity to the SDN controller, which sets up paths P3 and P4 (details are omitted in the workflow for simplicity). Once the NS has been reconfigured, the SO requests the MLFO to reconfigure the MAS pipeline and gives the details of the new forwarding graph (4). The MLFO computes the new MAS pipeline, which includes new Agent 3 and requests the SO its deployment (5) (see [8] for details). The SO deploys the container for the new agent with the help of the local virtual infrastructure manager in Site C, as well as the MAS pipeline connectivity with the help of the SDN controller (6). Then, the MLFO configures the new agent (7) that is ready to participate in the nRT control of the NS, which is communicated to the other agents in the MAS pipeline (8). Therefore, Agent 1 distributes the traffic model for the service to new Agent 3 (9), which uses it to select the RL model that needs to be used to operate the service. The procedure includes injecting traffic following the received traffic model through paths P3 and P4 to measure the delay (10). An algorithm takes such measurements and selects the model that better fits the new scenario from the once that are already pre-trained and available in the MLFO [6]. Finally, Agent 3 requests that model to the MLFO and it is ready to operate (11).

IV. Traffic Prediction and Model Selection

Knowledge sharing between agents is implemented for smooth NS reconfiguration. Specifically, Agent 1 shares with Agent 3 a traffic model for the NS, which has been trained in parallel to routing operation. Both traffic and routing operation models are based on Twin Delayed Deep Deterministic Policy Gradients (TD3) [2]. In the case of traffic prediction, the model predicts the NS traffic for a time window in advance. The state, reward, and actions are computed periodically. The state is a vector with the input traffic over the last period until t. Action is a vector with the minimum and maximum traffic expected during the period. The reward uses the maximum of differences between prediction and measured traffic (see eq. (1)).

$$reward = -n * \max(E_{min}, E_{\max}) \tag{1}$$

The used notation is as follows: *i)* M: set of models, each characterized by $<d_{max}$, P^i_{max}, P^i_{min} for every path considered by the model}, where P^i_{max}, P^i_{min} are the percentages of traffic sent through P^i; *ii)* S: set of probe scenarios, index s. Each scenario is characterized by $\{P^i_{(\cdot)}\}$; *iii)* D: set of delay measurements, $D=\{d_s, \forall s \in S\}$; *iv)* n: maximum traffic of the flow; *v)* $Tr_{(\cdot)}$: Actual min and max traffic for the window; *vi)* $Pr_{(\cdot)}$: Predicted min and max traffic for the window; and *vii)* $E_{(\cdot)}$: Absolute prediction error; abs($Tr_{(\cdot)}$ - $Pr_{(\cdot)}$). The traffic model is used for reproducing operation scenarios that will happen after the handover. To that end, the new agent uses the following to select a model able to ensure the required performance. The algorithm receives the set of models M, the set of probe scenarios S, and the set of delay measurements D. The models in M have been trained on specific topologies with unknown background traffic and validated in completely different topologies, which enables that the operation of each model can be characterized by attributes like delay and routing policies. The algorithm compares the delay measured under each scenario with d_{max}. If the maximum delay is guaranteed, the models that can operate in that probe scenario are selected as candidate models and labelled by distance to the measured delay. If there are candidate models, the one with the lowest distance is selected; otherwise, no model is selected, which is notified to the SO via the MLFO to provide more resources for the NS.

V. Illustrative Results

Model training and operation testing was carried out on a Python-based ad-hoc simulator. The scenario in Fig. 1a was used for training purposes to create a set of models M. For evaluation purposes, the operation is carried out on a different and more complex scenario, where P3 and P4 have 3 hops. Independent background traffic flows were used for loading the links, which are unknown to the models.

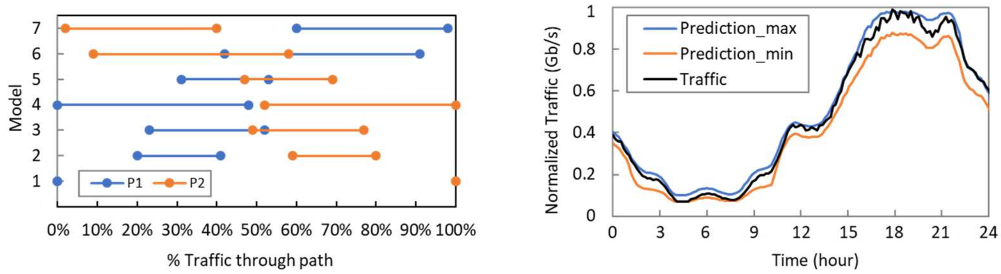

Fig. 2: Model behavior ranges for % of Traffic sent to P1 (a). Predicted and actual traffic (b)

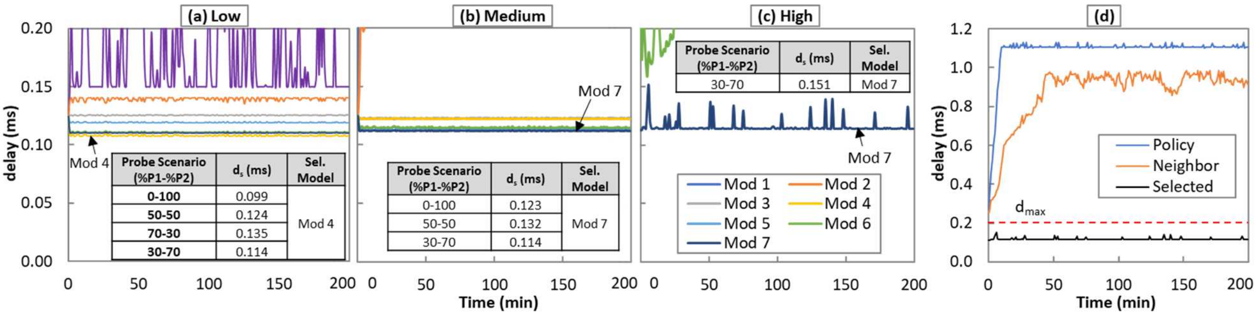

Fig. 3: Algorithm validation for (a) Low, (b) Medium, and (c) High NS traffic. (d) Handover performance.

Let us start with the characterization of the models. Fig. 2a presents the pretrained models and the range of percentage of traffic though P1 (Y-axis is for illustrative purposes and has no units). In this case, all the models work for two routes. We observe that with 7 models all possible routing percentages can be covered. Let us now evaluate the accuracy of the prediction model. Fig. 2b shows the results of the traffic prediction model for a period of one day, where estimation was made periodically every 10 min for a time window of 10 min ahead. We observe that min and max traffic predictions are very accurate and they estimate the actual traffic with a low margin, which validates the goodness of the predictor.

Armed with the NS traffic model, traffic prediction is used to actively probe P3 and P4 on the operation scenario. The algorithm described above was used for model selection and evaluation was carried out on three different NS traffic scenarios with low, medium and high traffic, assuming d_{max} = 0.2 ms. For each traffic scenario, several probe scenarios were used ranging from 0% to 100% of the NS traffic sent through P1. Fig. 3(a-c) plot the delay obtained when the NS is operated with each model. The inset tables show the probe scenarios that ensured d_{max}, along with the measured delay and the model selected by the algorithm. Under the low NS traffic scenario, several probing scenarios can ensure d_{max} (Fig. 3a) and the algorithm selects Model 4 as the most adequate for this scenario based on the measurements. This selection can be validated by running each model in the testing topology. We observe that although several models ensure d_{max}, Model 4 produces the lowest delay which validates the right selection. Under medium and high NS traffic, the probing scenarios, and thus, the number of candidate models that assure d_{max} reduces. Under medium NS traffic (Fig. 3b), the algorithm selects Model 7 and we observe in the plots that such model is the one providing better performance. Finally, under high NS traffic (Fig. 3c), only one proving scenario guarantees d_{max} and the algorithm selects Model 7. Note that in this traffic scenario, delay spikes are also obtained although d_{max} is not exceeded. Interestingly, although Model 6 shows large overlap with Model 7 in Fig. 2, it is not able to support d_{max} for the high traffic scenario. It is precisely under high traffic scenarios where the selection of the model to use for operation becomes critical in offering a smooth handover process.

Finally, Fig. 3d examines the performance after handover. Three benchmarking options are considered: 1) operation with the same model used by *neighbor* Agent 1; 2) a *policy* that distributes traffic evenly over all paths; and 3) the model *selected* by the algorithm. We observe that only the selected model assures d_{max}, which motivates the right model selection.

VI. Conclusions

A solution leveraging knowledge sharing and network probing to guarantee NS performance in the event of mobility scenario, has been proposed and validated. By implementing a MAS framework, agents are able share routing and traffic models when notified of an NS reconfiguration to intelligently guarantee an e2e delay service requirement. Experiments show that having a complete model data base is critical in high traffic scenarios to select the appropriate model and maintain an acceptable e2e delay after handover.

Acknowledgment

The research leading to these results has received funding from the EC thought the HORIZON SNS JU DESIRE6G (G.A. 101096466) and from the ICREA Institution.

References

[1] A. Bernal, M. Richart, M. Ruiz, A. Castro, and L. Velasco, "Near Real-Time Estimation of End-to-End Performance in Converged Fixed-Mobile Networks," Elsevier Computer Communications, vol. 150, pp. 393-404, 2020.

[2] S. Barzegar, M. Ruiz, and L. Velasco, "Packet Flow Capacity Autonomous Operation based on Reinforcement Learning," MDPI Sensors, vol. 21, pp. 8306, 2021.

[3] H. Shakespear-Miles, Q. Lin, S. Barzegar, M. Ruiz, X. Chen, L. Velasco, "Centralized and Distributed Approaches to Control Optical Point-to-Multipoint Systems Near-Real-Time," IEEE/OPTICA Journal of Optical Communications and Networking (JOCN), vol. 16, pp. 565-576, 2024.

[4] A. Dorri, S. Kanhere, and R. Jurdak, "Multi-agent systems: A survey," IEEE Access, vol. 6, pp. 28573-28593, 2018.

[5] L. Velasco, S. Barzegar, D. Sequeira, A. Ferrari, N. Costa, V. Curri, J. Pedro, A. Napoli, and M. Ruiz, "Autonomous and Energy Efficient Lightpath Operation based on Digital Subcarrier Multiplexing," IEEE Journal on Selected Areas in Communications (JSAC), vol. 39, pp. 2864-2877, 2021.

[6] S. Barzegar, M. Ruiz, and L. Velasco, "Autonomous Flow Routing for Near Real-Time Quality of Service Assurance," IEEE Transactions on Network and Service Management (TNSM), vol. 21, pp. 2504-2514, 2024.

[7] T. Rodrigues, S. Verma, Y. Kawamoto, N. Kato, M. Fouda, and M. Ismail, "Smart Handover with Predicted User Behavior using Convolutional NN for WiGig Systems," IEEE Network, vol. 38, pp. 190-196, 2024.

[8] P. Gonzalez, A. Zahir, Ch. Grasselli, A. Muñiz, M. Groshev, S. Barzegar, F. Callegati, D. Careglio, M. Ruiz, L. Velasco, "Deployment of Secure Machine Learning Pipelines for Near-Real-Time Control of 6G Network Services," in Proc. Optical Fiber Communication Conference (OFC), 2024.