European Conference on Optical Communication (ECOC) 2022 © Optica Publishing Group 2022

Multi-Agent-based Dynamic Optical Subcarrier Allocation for Near Real-Time P2MP Operation

H. Shakespear-Miles[1], M. Ruiz[1], A. Napoli[2], and L. Velasco[1*]

[1]Universitat Politècnica de Catalunya (UPC), Barcelona, Spain (*luis.velasco@upc.edu)
[2]Infinera, Munich, Germany

Abstract *We propose a multi-agent system (MAS) to manage subcarrier allocation in point-to-multipoint connectivity based on Digital Subcarrier Multiplexing. Similar performance to a centralized approach is shown, which allows for near-real time operation with increased scalability. ©2022 The Author(s)*

1. Introduction

Point to multi-point (P2MP) connectivity has shown great promise in 5G/6G scenarios [1]. Its deployment can also result in cost savings due to the reduction in the number of optical transponders to be equipped, when compared with point to point (P2P) connectivity; note that one single transponder installed in the hub node can serve several in the leaves [2]. In addition, the inherent ability of Digital Subcarrier Multiplexing (DSCM) optical systems to activate each subcarrier (SC) independently in the system in near real-time makes them ideal to adapt the capacity of the system to traffic dynamics, thus reducing energy consumption [3]. Coordination between transmitter (Tx) and receiver (Rx) without Software Defined Networking (SDN) controller intervention was also shown in [3].

P2MP connectivity, when supported by DSCM, leads to cost reduction in the presence of large dynamic traffic scenarios if not all SCs need to be active when traffic is low. In our previous work in [4], we proposed a centralized module running in the SDN controller to dynamically allocate SCs based on the traffic observed at the individual Txs in the leaves of a P2MP connection. Although this centralized approach successfully demonstrated the gains in the maximum number of leaves that can be supported, there were drawbacks: it is still heavily reliant on the SDN controller to communicate with the various Txs, it intakes large amounts of observational data, and it requires to process the requests on a synchronous basis, and all near real-time to follow traffic dynamics.

In this paper, we distribute decision making down to the very transponders participating in the P2MP connection and relieve the SDN controller from near real-time operation, hence increasing scalability. To this end, we introduce agents with the ability to communicate among them directly to create a *Multi-Agent System* (MAS). The target is to achieve gains similar to those provided by the centralized approach.

2. MAS-based Subcarrier Allocation

To show the main difficulty of moving decision making to the transponders, we focus on the direction from the leaves to the hub (MP2P). Here, some sort of coordination is needed to avoid two Txs using the same SC. Figure 1(a) illustrates the target scenario. Every Tx is tuned on the portion of the spectrum assigned (dotted lines), within which its SCs are allocated.

We define a normative MAS with two types of agents for the Txs and the Rx. There are various interpretations of normative MAS from which we adopt the definition in [5], where the system is governed by restrictions on patterns of behaviors of the agents in the system. In the proposed MAS, Tx and Rx agents have distinctive norms that govern their behaviors. The proposed algorithms rely on agent sociability, where agents can share knowledge to achieve their goals [6]; hence, communication is an important aspect of agents' functionality. We define communication channels capable of sending and receiving information between each Tx and the Rx, while avoiding Txs communicating with one another.

The Tx agent is responsible for allocating enough capacity for the incoming traffic. Traffic prediction is used to anticipate traffic dynamicity, and required capacity changes (activation or deactivation of SCs) are requested to the Rx agent. The role of the Rx agent is to mitigate SC oversubscription, which can occur when multiple Txs request the activation of a single SC. As for the Rx agent, we consider two main functionalities, used in combination: i) a *simple request process*, where a Tx agent makes a request and the Rx evaluates the spectrum in order to accept or deny this request. This is shown in Figure 1(b), where Tx1 sends a request for an increase in capacity. The Rx replies with instructions for the Tx to occupy SC 4. Another example is shown in Figure 1(c), where the Rx is aware of a possible oversubscription of SC 5 and commands Tx1 to use SC 1; and ii) a *neighboring Tx shifting* in order to satisfy a Tx's request. For example, in Figure 1(d), Tx2 requests additional

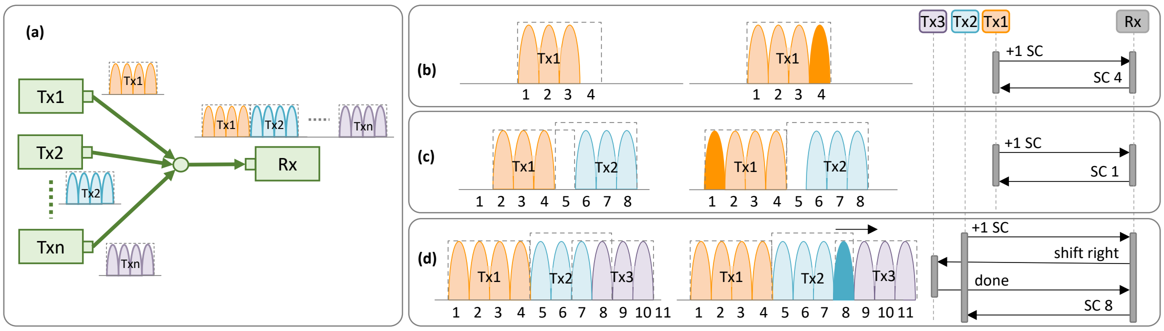

Figure 1: MP2P connectivity based on DSCM (a). SC allocation upon Tx1 request (b) and (c). Neighbor shifting (d).

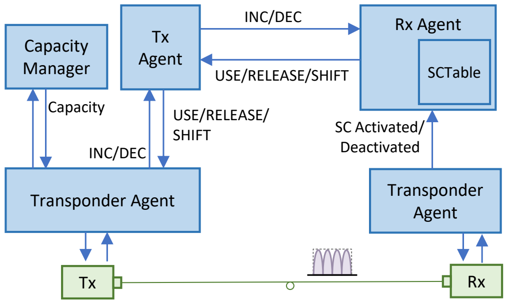

Figure 2: Relation between MAS and transponder agents

capacity, but there is no free spectrum since SCs 5 and 8 are already allocated. In that case, the Rx can ask the neighbor Tx3 to shift the SCs right and liberate SC 8, so that Tx2 can allocate it.

3. Architecture and Agent Models

In the previous section, Tx and Rx agents played several roles, from capacity and SC management to communication. However, the main role of MAS agents is to communicate with one another to coordinate SC utilization in the MP2P connection and avoid oversubscriptions. The MAS Tx agent also interacts with the local *capacity manager* which manages the capacity available between the local Tx and the Rx to follow traffic dynamicity. Tx and Rx MAS agents also interact with the *transponder agent* responsible for SC operation, such as activation and deactivation to meet capacity requirements, but it does not make any decision on the use of the spectrum. The spectrum management role is now played by the MAS Rx agent. Figure 2 summarizes the relationship among the different agents in the system and includes the main commands that they exchange.

The MAS Tx agent role is simply to translate requests to/from the transponder agent to/from the MAS Rx agent. However, the role of the MAS Rx agent is more complex as it has to coordinate spectrum allocation of the whole MP2P connection. Algorithm 1 describes the Rx agent; it receives a command and the ID of the Tx agent that issued it (we used the value -1 to indicate an invalid ID) and returns the command to be executed by a Tx agent on a specific SC.

The Rx agent processes messages from Tx

Algorithm 1. Rx Agent

```
INPUT: cmd, TxId | OUTPUT: cmd, TxId, SCId
1:  if cmd in {SCActivated, SCDeactivated} then
2:      <cmd, TxId> ← pendingReq.remove()
3:  if cmd == DEC then
4:      SCId ←find_best(SCTable, TxId, RELEASE)
5:      return {RELEASE, TxId, SCId}
6:  if cmd == INC then
7:      SCId ← find_best(SCTable, TxId, USE)
8:      if SCId <> -1 then return {USE, TxId, SCId}
9:      shift=<TxId, dir> ← find_neighborShift (
                 SCTable, TxId, SHIFT)
10:     if not shift then return {USE, TxId, -1}
11:     pendingReq.add(<cmd, TxId>)
12:     return {SHIFT, shift.TxId, shift.SCId}
```

agents to request or release a SC and from the local transponder agent when a SC has been actually activated or deactivated. Because operation is asynchronous, a list of requests pending to be processed is maintained. Pending actions are processed back after SCs are actually activated or deactivated (lines 1-2). In addition, the Rx agent maintains an internal table with the status of every SC, which is checked and updated when a request to release a SC or to get a free SC is received from a Tx agent.

When a request to release a SC is received (lines 3-5), the Rx agent finds which SC is the best to be released as a function of the allocation of neighboring Txs. The SC selected is returned to be sent to the requesting Tx agent. The allocation of a new SC entails more complexity (lines 6-12). If a neighboring SC to the Tx current allocation is free, then it is selected (lines 7-8). Otherwise, a possible spectrum shifting (entailing the activation and deactivation of two SCs) of a neighboring Tx is evaluated (lines 9-12). If a spectrum shifting is possible, it is requested to that Tx agent and the current SC allocation request is added to the pending list.

4. Results

In order to evaluate the proposed MAS system, a Python-based simulator was implemented to reproduce the MP2P optical connection in Figure 1 and include the agents and communication channels in Figure 2. Each

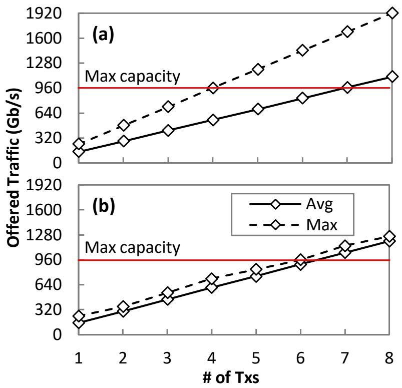

Figure 3: Offered Traffic at Rx

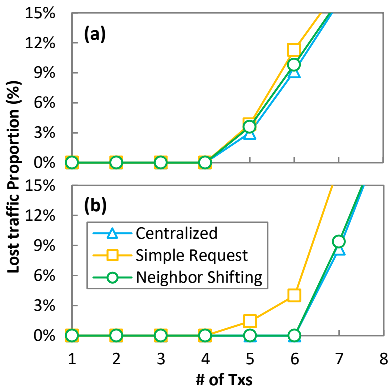

Figure 4: MAS vs Centralized

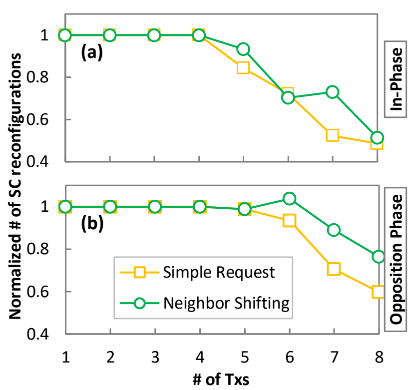

Figure 5: SC reconfigurations

Table 1. Message Exchange Analysis

#	In-phase		Opposition	
Tx	***avg(msg/Tx)***	***% shifting***	***avg(msg/Tx)***	***% shifting***
4	1.67	0%	1.78	0%
5	2.02	17%	1.86	6%
6	1.72	2%	2.35	24%

Tx was equipped with 4 60Gb/s SCs (assuming 16QAM at 11 Gbaud). Besides each Tx, a packet traffic generator was used to inject traffic following a typical daily profile varying between 60 and 240 Gb/s, thus leading to capacity requests between 1 and 4 SCs. On the Rx side, 16 SCs were configured, thus leading to a maximum capacity of 960 Gb/s for the whole MP2P connection.

Two traffic scenarios were considered, namely, *in-phase* and *opposition-phase*, with Txs requiring either a similar or different (respectively) number of SCs at a given time. Figure 3 shows the total offered traffic (average and maximum) as a function of the number of Txs. The in-phase scenario presents a high peak/average ratio (1.7) similar to that of a single Tx traffic. Thus, although 4 Txs produce moderate average traffic, the maximum reaches Rx capacity limit. On the other hand, as a consequence of traffic multiplexing, the opposition-phase scenario presents a much lower peak/average ratio (1.1), reaching maximum Rx capacity when 6 Tx are considered.

Two different configurations for the MAS Rx agent with increasing functionalities were evaluated: *i*) *Simple Request*, corresponding to the process illustrated in Figure 1b-c and defined by lines 1 to 8 in Algorithm 1; and *ii*) *Neighbor Shifting*, adding the process illustrated in Figure 1d to the previous functionalities and defined by the whole Algorithm 1. Figure 4 shows the performance of the MAS Rx agent configurations as a function of the number of Txs. For benchmarking purposes, a synchronous centralized approach based on the optimization model presented in [4] was implemented and executed every minute with monitoring data received from Txs. In light of Figure 4a, we can conclude that, under the in-phase scenario, both MAS configurations reached a similar performance to the centralized one, accepting 4 Tx without loss and 5 Txs with moderated loss ~3%. However, under the opposition-phase scenario (Figure 4b), the maximum of 6 Tx without loss is achieved by both centralized and MAS with neighbor shifting.

Complementing the previous results, Figure 5 shows the average number of SC reconfigurations per Tx for both traffic scenarios. Values are normalized to the number of reconfigurations with only one Tx. We observe that both methods perform the same number of SC reconfigurations when the spectrum is not at saturation; when the spectrum is near saturation, the MAS with neighbor shifting performs more reconfigurations to accommodate Tx requests.

Finally, Table 1 focuses on MAS with neighbor shifting and summarizes the average number of messages exchanged between a Tx and the Rx at a given time, as well as the percentage of those messages belonging to the neighbor shifting process. We observe that a moderated number of extra messages (up to 24% w.r.t that of the simple request process) are enough to eliminate loss and achieve near-optimal performance.

5. Conclusions

A MAS system for near real-time optical SC allocation has been presented. The distributed MAS reaches similar gains to centralized SC management. However, by moving decision making to the transponders in the P2MP connection a much more scalable solution can be created, thus relieving the SDN controller from operation after the provisioning phase.

Acknowledgements

This work has received funding from the EC through the H2020 B5G-OPEN (G.A. 101016663), the MICINN IBON (PID2020-114135RB-I00) project, and the ICREA Institution.

References

[1] D. Welch *et al*., "Point-to-Multipoint Optical Networks Using Coherent Digital Subcarriers," IEEE/OPTICA Journal of Lightwave Technology, vol. 39, pp. 5232-5247, 2021.

[2] M. Ruiz and L. Velasco, "Performance Evaluation of Light-tree Schemes in Flexgrid Optical Networks," IEEE Communications Letters, vol. 18, pp. 1731-1734, 2014.

[3] L. Velasco *et al*., "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.

[4] H. Shakespear-Miles, M. Ruiz, A. Napoli, and L. Velasco, "Dynamic Subcarrier Allocation for Multipoint-to-Point Optical Connectivity", International Conference on Photonics in Switching and Computing, 2022.

[5] G. Andrighetto, G. Governatori, P. Noriega, L. van der Torre, "Normative Muti-Agent Systems," in Dagstuhl follow-ups, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, vol. 4, 2013.

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