IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 November 2022, accepted 9 December 2022, date of publication 22 December 2022,
date of current version 29 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3231607

RESEARCH ARTICLE

Automatic Network Slicing for Admission Control, Routing, and Resource Allocation in Underwater Acoustic Communication Systems

OSAMA M. BUSHNAQ[1], (Member, IEEE), IGOR V. ZHILIN[1], (Member, IEEE), GIULIA DE MASI[1], (Senior Member, IEEE), ENRICO NATALIZIO[1,2], (Senior Member, IEEE), AND IAN F. AKYILDIZ[1], (Life Fellow, IEEE)

[1]Autonomous Robotics Research Center (ARRC), Technology Innovation Institute (TII), Abu Dhabi, United Arab Emirates
[2]Université de Lorraine, CNRS, LORIA, F-54000 Nancy, France

Corresponding author: Osama M. Bushnaq (osama.bushnaq@tii.ae)

This work was supported by the Technology Innovation Institute.

ABSTRACT Large-scale underwater wireless communications networks are gaining popularity because they are a critical enabler for a variety of applications in the environmental, commercial, and civilian domains. Multiple applications with various service-level-agreement (SLA) requirements can be provided utilizing the same network infrastructure to enable cost-effective underwater (UW) network deployment and maintenance, allowing for globally efficient resource management. Because various stakeholders may have different service level agreements (SLAs), underwater acoustic communication systems (UACS) must provide functional isolation of services. As a result, network slicing is critical in such networks. In this paper, a novel optimization framework for automated network slicing (ANS) in UACS is presented, which enables SLA-based admission control, routing, and dynamic resource allocation. It achieves optimized solutions and improves network performance, decreases deployment costs and simplifies network operation This paradigm is also in line with the development of 5G/6G networks. The proposed automatic slicing framework considers the difficult underwater acoustic channel characteristics to deliver a heuristic sub-optimal routing and resource allocation solution based on the SLA provided/required by network tenants/applications. An in-depth numerical analysis is used to test the suggested solution and compare it to state-of-the-art software-defined networks (SDN) routing and resource allocation algorithms.

INDEX TERMS Underwater acoustic communication systems, automatic network slicing, software-defined networks, routing, resource allocation.

I. INTRODUCTION

Wireless communications and networking in the undersea environment are highly difficult. The fundamental reason is due to the peculiarities of communication channels in an underwater environment. In salty water, radio frequency (RF) and magnetic induction (MI) are heavily absorbed, limiting transmission distance to ∼100m at frequencies ∼ 30Hz−300Hz [3], [4]. Due to turbulence in the underwater environment, optical communications are affected by large scattering losses, limiting their reliability and transmission distance. [5], [6], [7], [8]. Despite its propagation latency and multi-path fading, acoustic communications technology is the most common physical layer technology in underwater networks. Underwater acoustic communication (UAC) is a long-range, dependable solution for low-rate wireless communication in a variety of water conditions [9], [10], [11], [12], [13], [14], [15]. Other problems of underwater network realization include high cost, low level of supervision, and limited power supply, in addition to the harsh communication characteristics. As a result, for future underwater communication networks, efficient resource allocation

The associate editor coordinating the review of this manuscript and approving it for publication was Jiajia Jiang.

and cost-effective solutions are critical. In order to provide efficient, dependable, and adaptable underwater acoustic communication systems (UACS), novel and specialized networking technologies are required [16]. [11], [12], [13] discuss techniques for improving acoustic link performance, [17], [18], [19] discuss the issues of underwater network protocol stack layers and recent design solutions. In [20], the authors presented a space-frequency division multiple access (SFDMA) scheme for a MIMO supported UACS. The SFDMA scheme in [20] considerably improves the node access percentage. In [21], a throughput maximization algorithm is proposed for UAC network with energy harvesting (EH) capabilities. The commercial and research state-of-the-art underwater acoustic modems are reviewed in [22], where the current capabilities of such modems are comprehensively presented. Multi-mode software-defined radio implementing optical and magnetic induction paradigms was recently proposed in [23] to mitigate for acoustic communication weakness. Reference [24], [25] presented Software Defined Networks (SDN) as a way for implementing large-scale UACS. In [26] and [27], innovative AUV-assisted routing solutions were proposed to improve underwater network connectivity. In [26], a fuzzy-logic-based algorithm is used with AUV trajectory re-planning to improve network connectivity. In [27], a particle swarm optimization algorithm and Markov chain model are utilized to optimize node deployment by the AUV to enhance data flow over the network. These solutions require AUVs in the first place. Moreover, the reliance on AUV might introduce extra delay unless the data communication is predictable. A cooperative routing protocol is proposed in [28] for energy-efficient underwater acoustic communication, where data can be re-transmitted from relay nodes. In [29], an ant-colony based cooperative routing is proposed to balance payload over energy-limited underwater network nodes. In [28] and [29], all nodes have information about other nodes' locations or link states, which requires high overload to maintain. Furthermore, the aforementioned routing solutions are designed for specific applications. In the proposed ANS solution, the global view of the network is only given to the SDN controller deployed on the water surface, where routing and resource allocation decisions are made. Based on the application requirements and priority, network resources are shared optimally. Finally, thanks to the network slicing based solution concept, isolation between applications/slices and performance guarantees are maintained in our ANS solution.

The SDN architecture allows for the separation of the control and data planes, performing globally optimized resource management at a controller node, which acts as the network brain. Moreover, high flexibility, programmability, and network architecture virtualization are other advantages that improve network performance in terms of resource utilization, network management, operating costs, enabling innovation and evolution.

To enable efficient resource management on top of SDN architecture, network slicing (NS) can be utilized. Consider a network serving multiple applications or agents, where the connectivity requirements are different per application/agent. For example, one UACS network could be used to provide underwater telemetry (delay tolerant and requires low data rate), underwater emergency communication (delay intolerant but requires low data rate), underwater imaging and surveillance (delay tolerant but requires high data rate), and so on. NS offers a cost-effective and bandwidth-efficient solution to serve multiple UACS applications/agents with different Service Level Agreement (SLA) requirements, using the same network infrastructure. This is done by creating multiple network slices on top of a common shared physical infrastructure, where a network slice is a logical network that provides specific network capabilities and characteristics in order to serve a defined business purpose of a customer [30]. As a result, network resources are provided as per application (i.e. slice) requirements. Furthermore, network slicing naturally provides functional isolation of services across multiple agents. Hence, the cost is significantly reduced, since the same network infrastructure can be shared among multiple agents. For these reasons, we propose an UACS network slicing solution. To the best of our knowledge, our paper is the first work to consider network slicing for underwater acoustic communication networks.

Network slicing has been extensively studied for next-generation cellular networks, i.e., 5G/6G [31], [32], [33], [34]. Resource allocation, including spectral bandwidth, transmission power and cache storage for NS plays a pivotal role in load balancing, resource utilization, and networking performance. Several principles and schemes of resource allocation algorithms in 5G NS are surveyed in [26]. In [35], network slicing is performed on radio access network (RAN) bandwidth resources, base-station (BS) cache storage, and backhaul capacity. In [36], NS in heterogeneous-cloud RAN (H-CRAN) network is performed as a process of allocating network resources to users associated with different tenants/slices. The authors in [36] performed a slicing scheme that consists of an upper-level slicing, which manages admission control, user association, and baseband resource allocation; and a lower-level slicing, which performs radio resource allocation among users. In [37], a service-oriented deployment policy of end-to-end 5G network slicing is proposed, where slice requests are mapped to network infrastructure. Automatic network slicing (ANS), where network slicing is purely SLA-based and does not require extra knowledge of the resource requirements associated with a slice, was recently proposed for 5G [38], [39], and for low Earth orbit (LEO) satellite assisted networks in [40]. In ANS, tenants no longer need to model their slices in terms of explicit resource requirements. Instead, slices' flows are created, admitted, and managed in ANS automatically based on the SLA, where a *flow* is a direct or indirect connection between a source and a destination node to serve a given slice. Such an SLA-based slicing can help abstract the complexities of slice implementation and customization to address different use cases, and slice priorities [40]. We extend our presentation

in [1], by considering in this paper a more realistic system model that takes into account the network evolution over time by setting a reservation period for each admitted network flow. Consequently, only part of the previously admitted network flows is released at each slice evaluation period, while new flows are admitted depending on the remaining unoccupied network resources. Moreover, improvements to the user satisfaction modeling and solution algorithms have been implemented. Finally, more insights and numerical results are obtained to clarify the impact of adopting network slicing solutions for future UACSs.

In this paper, we propose a novel optimization framework for ANS in UACS. The ANS solution significantly reduces the overall network deployment cost per agent and enhances the quality of service (QoS) and resource management by achieving the SLA requirements while allocating minimum network resources. Such paradigm is also compatible with terrestrial networks, i.e., 5G/6G networks, hence, hybrid terrestrial-underwater connectivity is simplified. Because of the differences in channel characteristics, problem limitations, and network architectural complexity, applying 5G/6G slicing methods to UACS is prohibitive. The suggested automatic slicing architecture has been specifically designed to accommodate the difficult underwater acoustic channel characteristics. For instance, acoustic channel path loss is a function of transmission frequency and distance, creating long-range and mid-range transmission bands depending on the transmission frequency. Taking these transmission bands' features into account at the network management level is critical for optimal resource allocation. Additionally, underwater acoustic propagation delay is extremely high. Hence, routes that propagate over long distances can be quickly eliminated, which reduces the computational complexity of ANS. Furthermore, the proposed ANS solution takes advantage of the simplified small to mid-size underwater network architecture, where no network abstraction is needed. In short, the underwater ANS solution must consider differences between the terrestrial radio and underwater acoustic channel characteristics, network architecture, and problem constraints. More details of the differences between underwater acoustic and terrestrial radio networks are discussed in [18] and [41].

Finally, underwater nodes' energy and computational resources are extremely scarce. Since our ANS solution relies on SDN network architecture, most of the computations (including route selection, and resource allocation) are managed by the controller. Having the controller placed on the sea surface, higher energy and computation resources are affordable (for example, it is possible to have solar/wind energy harvesting, periodic recharging, high-speed computer deployment, etc.). Therefore, one of the important advantages of our proposed network solution is the reduction of energy and computation at the underwater nodes. A novel systematic flow admission control, routing, and resource allocation solution is offered and numerically assessed using a thorough use-case-driven assessment scenario based on the SLA given/required by the tenants/applications. Furthermore, we compare the network performance with recent routing and resource allocation schemes for vanilla-SDN[1] [42]. Although concepts such as SDN and NFV have been suggested for underwater networks in [24], [43], and [44], none of these works provided solutions for routing and resource allocation. The contribution of this work is summarized as follows:

- To the best of our knowledge, our paper is the first work considering the automatic network slicing (ANS) problem in acoustic underwater communication networks. As known from the wireless and wired terrestrial networks, ANS allows for better resource management, performance guarantees for each agent, pricing per connectivity requirements (i.e., SLA requirements), and isolation from other types of traffic.
- A novel ANS model for the UACS is proposed. Based on an SDN architecture with one controller on the sea surface, a centralized optimization algorithm is run at the controller to maximize a utility function that models ANS performance gain. Practical slice flow request management, admission, and release are presented.
- Sub-optimal heuristic solution to maximize performance utility is presented, where tenants are assigned network resources by direct user admission control, flow routing, and resource allocation. The solution is specifically tailored for the challenging UACS channel characteristics, where spectral bandwidth is limited and transmission delay is high [2].
- Comprehensive use-case-driven numerical evaluations are conducted to verify the robustness of the ANS solution. Furthermore, we compare the network performance with state-of-the-art routing and resource allocation schemes for a vanilla-SDN algorithm.

In the rest of the paper, the UAC channel is modeled in Section II, while we discuss the network architecture and theoretical problem formulation in Section III. A heuristic solution for the automatic slicing problem is proposed in Section IV, whereas finally the numerical analysis is provided in Section V, before we conclude the paper in Section VI.

II. UNDERWATER ACOUSTIC COMMUNICATIONS CHANNEL

Let us discuss the Underwater Acoustic Channel (UAC) properties that play a key role in the operation of the ANS for UACS. The UAC provides kilometer range connectivity at moderate to low data rates ($\sim$ 10 kbps or less) [9]. To tune the ANS specifically for the UAC, we use the following channel path loss (PL) model expressed in dB as [45],

$$\mathrm{PL}(f, d) = \kappa \log(d) + \alpha(f) d, \tag{1}$$

where κ is the spreading factor ($\kappa = 10$ for cylindrical transmission, $\kappa = 20$ for spherical transmission), d is the

[1]Vanilla-SDN is a simple SDN network that does not apply advanced network customization and intelligence such as network slicing and artificial intelligence (AI).

transmission distance, and $\alpha(f)$ is the Thorp's model absorption coefficient [41]. Various phenomena affect the channel performance: the multipath propagation, the increase of absorption for higher frequencies [9], and the noise power spectral density [41], [45]. From [9] we can see that the optimal frequency band depends on the transmission range and that the transmission bandwidth decreases with the communication distance increase. Thus, the multi-hop transmission may be more energy efficient. The signal-to-noise ratio (SNR) is expressed as,

$$\gamma = P_{\text{ac}} - \text{PL} - N, \tag{2}$$

where N is the noise level, and P_{ac} is the power spectral density (PSD) of the acoustic transmitted signal in dB re $\mu Pa/\sqrt{Hz}$ @ 1 m [45],

$$P_{\text{ac}} = P_{\text{elec}} + 170.8 + 10\log(\eta_{\text{e/s}}). \tag{3}$$

where P_{elec} is the electric power, and $\eta_{\text{e/s}}$ denotes conversion efficiency. The link capacity is expressed by the Shannon theorem as [45]:

$$C(\gamma) = \int_{BW} \log_2(1 + 10^{\gamma/10})df. \tag{4}$$

Fig. 1 shows the channel capacities for different transmission powers and frequency bands versus communication distance. As the distance increases, low-frequency transmission becomes more efficient. To model the imperfectness of the modulation and coding scheme (MCS), we add a performance gap of η dB relative to the Shannon limit. As a result, the communication rate is expressed as,

$$R = C(\gamma - \eta), \tag{5}$$

where we put $\eta \in (0, 10)$ dB for low multipath effects (in deep water) and $\eta \in (5, 20)$ dB for strong multipath in shallow water ([46], Ch. 6).

We do not model the dependence of the speed of sound on various parameters and set it to $c = 1500$ m/s. Thus, the transmission delay equals:

$$T = \frac{d}{c} + \frac{b}{R}, \tag{6}$$

where b is the packet size in bits.

III. NETWORK ARCHITECTURE MODEL AND PROBLEM FORMULATION

Fig. 2 displays the network architecture. The UACS duty is to offer the required communication quality of services (QoS) for a variety of applications, e.g., ecological sensing, underwater imaging and detection, etc. Two acoustic frequency bands, ($k \in K = \{k_1, k_2\}$) are used by the network's nodes to communicate. Each of these bands naturally has certain distinct characteristics, i.e., they might offer various connection quality-distance trade-offs. Take the 30 kHz and 300 kHz frequency ranges as an illustration. At short distances (up to 100 meters), the 300 kHz band may deliver 100s of kbps, while at long distances, the 30 kHz frequency can deliver several kbps data rates.

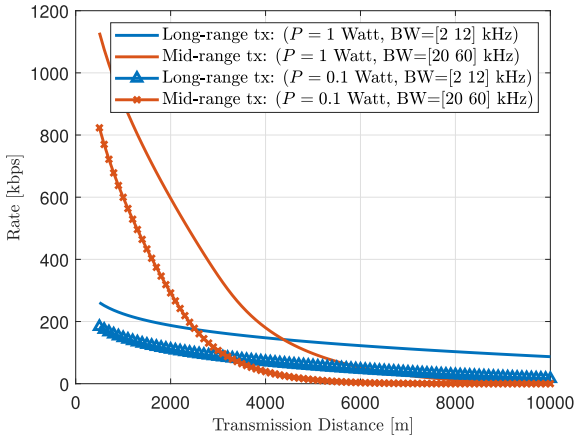


FIGURE 1. Underwater acoustic link capacity against transmission distance.

We consider an SDN network with one controller that is deployed on the water surface. We assume that the controller has a global view of the network status by periodically collecting information, such as node locations, channel state information (CSI), traffic, link loss/recovery, etc. Based on collected information, and required communication services, the controller makes decisions on admission control, routing, and resource allocation to different flow requests. Among other tasks, the controller is responsible for mobility management, session establishment, traffic management, and network virtualization. In addition to UACS management, the controller acts as a gateway connecting underwater data traffic to terrestrial networks. A network slice could be deployed over multiple operators and span across multiple parts of the network [30]. Therefore, a slice may comprise shared resources from the underwater network and other networks over the sea surface such as aerial and cellular networks, connecting end-to-end terminals. We focus our discussion on resource slicing and allocation for underwater networks. Although shared resources include processing/transmission power, memory storage, and spectral bandwidth, we focus on optimizing bandwidth (resource block) allocation and slicing, as it is the main limiting factor for UACSs.

Network capabilities, such as data rate, latency, reliability, and security, are provided to the network tenants/users based on SLA [30]. In this paper, the transmission latency and rate are optimized with constraints on the transmission power and bandwidth. Link reliability is considered inherently through modulation and coding scheme parameters as in (5).

To build a network that can automatically serve multiple UW communication and sensing applications, the network resources must be managed based on high-level SLA requirements. For each network slice flow s, an application/slice specifies a set of performance metrics that should be maintained for a transmission from a source node v_{src} to a destination node v_{dst}. The SLA requirements consist of slice priority $w_s \in [0, 1]$, target and threshold rates (R_s^{t}, R_s^{th}), target and threshold time delays (T_s^{t}, T_s^{th}), and weights for the rate and time delay significance ($w_{s,r} \in [0, 1]$, $w_{s,t} = 1 - w_{s,r}$).

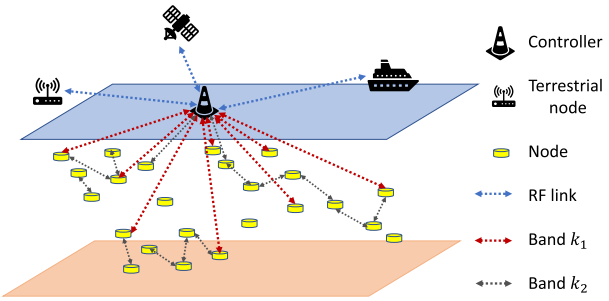


FIGURE 2. Underwater acoustic network architecture.

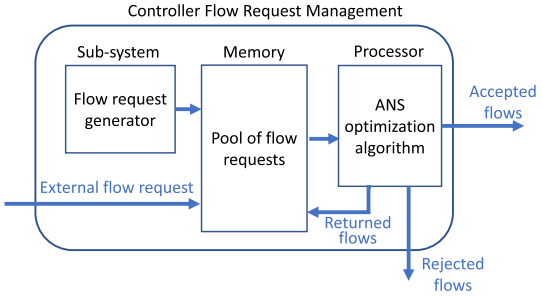


FIGURE 3. Slice request management at the controller.

As shown in Fig. 3, a flow request is initiated or received at the controller, to select a routing path and allocate network resources for that flow. A flow request for automatic, predictable, or periodic traffic [47] may be generated by the controller. Typically the flow requests are externally sent through the in-band control channels. The flow request can be generated at the source node or by other nodes (possibly over a terrestrial network) to request data collection. Assuming independent flow requests, the number of new requests over the *slice evaluation period* T_0 is modeled using a Poisson distribution with an average number of requests, λ. At the end of the slice evaluation period, the controller treats all the flow requests that are stored in its memory by running the ANS optimization algorithm. The treated flow requests are:

1) accepted, and given routing path and resources,
2) returned to the controller memory to be re-evaluated at the next slice evaluation period, if a threshold slice waiting time is not exceeded; or
3) rejected, if the threshold slice waiting time is exceeded after waiting for one or more slice evaluation periods. A warning message with the rejected slice SLA can be sent to the network management unit to indicate flow request rejection. This can help distinguish slices with SLA higher than the capacity of the network and notify higher network layers (i.e. application layer) to request slice re-transmission after some time depending on the network traffic.

Accepted slices' flows are given network resources according to the ANS algorithm for a period of time, denoted as T^s_{rsrv}. The *reserve period* equals to multiple slice evaluation periods and can be different for different flow requests, i.e., $T^s_{rsrv} = m_s T_0$ where $m_s \in \mathbb{Z}$ is a number of reserved slice evaluation periods. Therefore, the average amount of active data flows over the network can be approximated as,

$$\sum_{s=1}^{S} m_s A_s \tag{7}$$

where S is the number of required slices' flows with an average λ, and A_s indicates flow admission. The slice evaluation period, T_0 should be short to guarantee acceptable resource allocation delay, but long enough to allow joint route evaluation over a large number of flow requests. The objective of the ANS algorithm is to select the optimal routing paths and allocate network resources in a way that maximizes the number of served flows while offering a service that is as close as possible to the target requirements, but also within the threshold SLA requirements. Let us define a utility function that its maximization achieves the ANS objective:

$$U = \sum_{s} A_s w_s u_s, \tag{8}$$

where,

$$u_s = w_{s,r} u_{s,r} + w_{s,t} u_{s,t}, \tag{9}$$

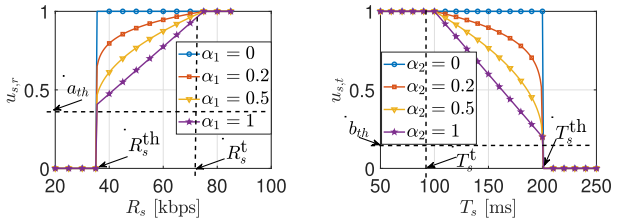
where, $A_s \in \{0, 1\}$ indicates whether the flow s is admitted, $A_s = 1$, or rejected with $A_s = 0$, and u_s represents the flow utility, which is a weighted sum of the flow rate utility $u_{s,r}$ and and flow delay utility $u_{s,t}$. Naturally, the satisfaction level is sub-modular with the flow rate and time-delay performance. Therefore, to practically reflect the satisfaction level in the utility function, concave non-decreasing functions are used to model the flow rate utility, and concave non-increasing functions are used to model the flow time-delay utility, i.e., [48],

$$u_{s,r} = \begin{cases} 0 & R_s \le R^{\text{th}}_s \\ a_{\text{th}} + (1 - a_{\text{th}}) \left(\dfrac{R_s - R^{\text{th}}_s}{R^{\text{t}}_s - R^{\text{th}}_s} \right)^{\alpha_1} & R^{\text{th}}_s \le R_s \le R^{\text{t}}_s \\ 1 & R^{\text{t}}_s \le R_s, \end{cases} \tag{10}$$

$$u_{s,t} = \begin{cases} 0 & T^{\text{th}}_s \le T_s \\ b_{\text{th}} + (1 - a_{\text{th}}) \left(\dfrac{T^{\text{th}}_s - T_s}{T^{\text{th}}_s - T^{\text{t}}_s} \right)^{\alpha_2} & T^{\text{t}}_s \le T_s \le T^{\text{th}}_s \\ 1 & T_s \le T^{\text{t}}_s, \end{cases} \tag{11}$$

where, R_s and T_s denote the flow rate and delay, which depend on the selected route and allocated resources to the flow s. The parameters a_{th}, b_{th}, α_1, and α_2 are given in the SLA to determine the threshold satisfaction and concavity of the curves, as shown in Fig. 4.

The UACS can be described using a graph, $\mathcal{G}(\boldsymbol{\nu}, \boldsymbol{\mathcal{E}})$, where $\nu \in \boldsymbol{\nu}$ represents a node and $\mathcal{E}_{\nu,\nu',k} \in \boldsymbol{\mathcal{E}}$ represents a link from ν to ν' over the spectral band $k \in K$. Let us consider a flow s that has a route from the source node $\nu^{(s)}_0 = \nu^{(s)}_{src}$ through a sequence of intermediate nodes $\nu^{(s)}_1 \dots \nu^{(s)}_{h-1}$ to the destination node $\nu^{(s)}_h = \nu^{(s)}_{dst}$, where $\{\nu^{(s)}_j\}, j \in \{0, \cdots, h^(s)\}$ is a set of all visited nodes in the flow s.The index indicates the j-th hop of the flow (s). Assuming the controller is aware of the network



(a) Flow rate utility function. (b) Flow delay utility function.

FIGURE 4. Concave flow rate and time-delay utility functions.

nodes locations the flow rate R_s and delay T_s can be obtained as:

$$R_s = \min_{j\in\{1,\cdots,h\}} \sum_{k\in K} R_{v_{j-1},v_j,k}\, x^{(s)}_{v_{j-1},v_j,k}, \tag{12}$$

$$T_s = \sum_{j\in\{1,\cdots,h\}} T_{v_{j-1},v_j}, \tag{13}$$

where $x^{(s)}_{v_{j-1},v_j,k} \in [0, 1]$ represents link's resource allocation given to the flow s, and $R_{v_{j-1},v_j,k}$ and $T_{v_{j-1},v_j,k}$ are the link's capacity and delay, as expressed in (5) and (6). In (12), we sum over the available communication bands between two links to allow communication over two bands simultaneously. Hence, the data rate is summed between any two nodes. Then, the link with the minimum available capacity determines the flow rate as it represents the route bottleneck. In (13), the delay is expressed as the summation of each traveled link delay.

For any flow, a node should not be revisited, hence,

$$\forall j \neq j' : v_j \neq v_{j'} \tag{14}$$

To model the limitation of link capacity and node power consumption, we define the following constraints

$$\forall v, v', k : \sum_s x^{(s)}_{v,v',k} \leq 1, \tag{15}$$

$$\forall v : \sum_{s,k} P^{(s)}_{v,k} = P_v \leq P_{\max}. \tag{16}$$

where $P^{(s)}_{v,k}$ is the transmission power per node, band and slice. P_v and $P_{\max}$ represent the total transmission power per node and maximum allowed transmission power per node, respectively.

To reserve the network's resources, only the flows that meet the slice SLA should be admitted and given access to the system resources, this is guaranteed if the following constraints are met,

$$R_s \geq A_s R_s^{\text{th}}, \tag{17}$$

$$A_s T_s \leq T_s^{\text{th}}. \tag{18}$$

We formalize mathematically the automatic network slicing (ANS) problem as follows:

$$\begin{aligned} \max_{A_s,\{v_j^{(s)}\},x^{(s)}_{v_{j-1},v_j,k}} \quad & U, \\ \text{subject to:} \quad & (14)-(18), \\ & A_s \in \{0, 1\}, x^{(s)}_{v_{j-1},v_j,k} \in [0, 1]. \end{aligned} \tag{19}$$

The described ANS problem in (19) cannot be solved directly since it is a *mixed integer non-linear program (MINLP)*. To reduce the complexity, we propose in the next section a sub-optimal solution in which the problem is decoupled into two sub-problems.

IV. NETWORK AUTOMATIC SLICING SUB-OPTIMAL SOLUTION

In this section, we propose a systematic solution that decouples the NP-hard problem in (19) into two solvable sub-problems. First, a route nomination and elimination algorithm is created to eliminate all the routes that do not obtain the minimum SLA requirements. Then, a route selection and resource allocation algorithm is created to select one of the nominated routes in the first sub-problem and allocate a suitable percentage of the involved links capacities.

A. ROUTES NOMINATION AND ELIMINATION

All routes that obtain the threshold SLA requirements for each flow are nominated as candidate routing solutions in this sub-problem, and all routes that fail to meet the ANS problem restrictions are discarded. At this point, each flow's pathways are nominated separately. The next subsection deals with joint route selection from nominated routes and resource allocation.

The overall propagation distance rises, and consequently the propagation delay also increases by adding more transmission hops as opposed to direct transmission from any node to the target node. However, according to the UWA channel characteristics outlined in Section II, limiting the number of transmission hops reduces the achievable rates per link and total traffic transferred over the network by increasing transmission distances per hop. As a result, slices that require high data rates should have a higher number of hops, whereas slices with a low time-delay SLA need should have a low number of hops.

The following is a description of the route nomination procedure for each flow request. To begin, we will make the three tables as follows:

- A table containing next-hop link rate and delay for each node v is created and sorted by link capacity, as Table 1 shows. In The table, $R_{v,v',k}$ and $T_{v,v',k}$ are determined as in (5) and (6), respectively.
- At each hop count, a table of incomplete candidate routes is constructed as illustrated as in Table 2 where R^r and T^r are the incomplete route's rate and delay, and $R_s^{\rightarrow v_{dst}}$ and $T_s^{\rightarrow v_{dst}}$ represent the overall flow rate and delay, given that data is transmitted directly to the destination after hopping through the corresponding incomplete route. According to rules for route elimination and nomination, as outlined below, Table 2 entries are filled subsequently starting with the first hop incomplete routes tables to a maximum number of hops, H incomplete routes table. The total number of hops is dependent on the network size. A small H value can reduce the

TABLE 1. Next-hop link rate and delay.

$\mathcal{E}_{\nu,\nu',k}$	$R_{\nu,\nu',k}$	$T_{\nu,\nu',k}$
⋮	⋮	⋮

TABLE 2. Hop h incomplete routes.

Route	R^r	T^r	$R_s^{\rightarrow v_{dst}}$	$T_s^{\rightarrow v_{dst}}$
⋮	⋮	⋮	⋮	⋮

TABLE 3. Nominated routes.

Route	R_s	T_s
⋮	⋮	⋮

solution complexity but may lead to a suboptimal solution. In our simulation, $H = 10$ is utilized as a backup convergence guarantee to terminate the algorithm.

- As illustrated in Table 3, a list of nominated routes per flow is produced.

In order to automatically eliminate routes that do not obtain the minimum required slices' SLA, a set of rules are followed. (1) Any visited node in the unfinished route cannot be revisited. (2) The neighbors in the next hop are those with $R_{\nu,\nu',k} \geq R_s^{\text{th}}$, $\forall \nu \neq \nu'$. All paths to nodes that are not neighbors are deleted. (3) If $T_s^{v_{dst}} \geq T_s^{th}$, the incomplete route is removed because it fails to meet the slice threshold time-delay requirements. This rule immediately eliminates routes that travel in the opposite direction as the target node. (4) The route is added to the database of nominated routes if $R_s^{\rightarrow v_{dst}}$ and $T_s^{\rightarrow v_{dst}}$ meet the required slice's SLA.

Algorithm 1 summarizes the procedure followed to fill the nominated routes table for each flow, given links' capacity, SLA requirements of all slices, and percentage of traffic over all links by previously allocated flows. We start by updating the content of the $h = 1$ incomplete routes table by filling $v_{src} \rightarrow$ in the first column, indicating that the flow starts form the source node. At the beginning, $T^r = 0$ and $R^r = \infty$, since no transmission has taken place yet. The values of $R_s^{\rightarrow v_{dst}}$ and $T_s^{\rightarrow v_{dst}}$ in the incomplete routes table are calculated as $R_s^{\rightarrow v_{dst}} = \min\{R^r, R_{l,v_{dst},k}\}$ and $T_s^{\rightarrow v_{dst}} = T^r + T_{l,v_{dst},k}$ with l denoting the last visited node and $k \in K$. An entry is added to the nominated routes table only if $R_s^{\rightarrow v_{dst}} \geq R_s^{th}$ and $T_s^{\rightarrow v_{dst}} \leq T_s^{th}$, where the destination node v_{dst} is concatenated to the incomplete route. If on the other hand, $T_s^{\rightarrow v_{dst}} \geq T_s^{th}$, the incomplete route is directly eliminated from Table 2, as the threshold slice SLA time delay is exceeded.

For the transmission hops $h \in \{2, \cdots, H\}$, the incomplete routes tables are updated subsequently from $h = 2$ until $h = H$. At the route column, a new node is concatenated to the incomplete routes of the previous (i.e., $h-1$) incomplete routes table towards each neighbor of the last node in the hop $(h-1)$ incomplete route. The new route rate and time delay are calculated as in (12) and (13). Hence, $R^{r,h} = \min\{R^{r,h-1}, R_{\nu,\nu',k}\}$ and $T^{r,h} = T^{r,h-1} + T_{\nu,\nu',k}$. The steps above are The procedure is repeated until the hop (H) incomplete routes table is filled or until no more incomplete routes satisfy the SLA requirements. The operation above is continued until we update all the (H) incomplete routes tables or until there are no more incomplete routes that meet the slices' SLA.

Algorithm 1 Flow Nominated Routes

1: Fill the "first hop incomplete routes" table with:

Route	R^r	T^r	$R_s^{\rightarrow v_{dst}}$	$T_s^{\rightarrow v_{dst}}$
$v_{src} \rightarrow$	∞	0	R_{v_{src},v_{dst},k_1}	T_{v_{src},v_{dst},k_1}
$v_{src} \rightarrow$	∞	0	R_{v_{src},v_{dst},k_2}	T_{v_{src},v_{dst},k_2}

2: **for** $h = 2 : H$ **do**
3: **for** all "$(h-1)$-th incomplete routes" table rows **do**
4: **if** $R_s^{\rightarrow v_{dst}} \geq R_s^{th}$ and $T_s^{\rightarrow v_{dst}} \leq T_s^{th}$ **then**
5: add route to the "nominated routes" table
6: **end if**
7: **if** $T_s^{\rightarrow v_{dst}} \leq T_s^{th}$ **then**
8: Add a new route to the hop h incomplete routes table by adding a new hop towards each neighbor of the last node in the $(h-1)$-th incomplete route
9: Calculate $R^{r,h} = \min\{R^{r,h-1}, R_{\nu,\nu',k}\}$ and $T^{r,h} = T^{r,h-1} + T_{\nu,\nu',k}$
10: Calculate $R_s^{\rightarrow v_{dst}} = \min\{R^r, R_{l,v_{dst},k}\}$ and $T_s^{\rightarrow v_{dst}} = T^r + T_{l,v_{dst},k}$
11: **end if**
12: **end for**
13: **end for**

The table of nominated routes is filled using the technique described in this subsection, for each network flow. It is worth noting that an optimal solution can be obtained if the joint route selection and resource allocation in the next sub-problem is optimal. However, only a sub-optimal approach for the joint route selection and resource allocation is described in the following subsection.

B. ROUTE SELECTION AND RESOURCE ALLOCATION

For each data flow, a list of nominated routes was obtained in the previous subsection. Based on these lists, we first decide whether each flow should be admitted or rejected. If the flow is admitted, we allocate a percentage of the links' resource of one of the nominated routes, to maximize the utility function in (8). At this point, flow admission, route choice and resource allocation should be jointly conducted over all network's flow requests. However, to enable fast and dynamic resource allocation, the resource allocation per flow is done iteratively, depending on the slice priority, *ws*. Joint optimization is accomplished by considering the links capacity degradation due to previously admitted slices and selecting routes that are likely to be less congested by un-served flows. A flow is reviewed randomly at each iteration to be

able to eventually serve low-priority slices. The probability of flow evaluation at a given iteration is given as

$$P_s = \frac{w_s}{\sum_{i\in\mathcal{S}} w_i}, \ \forall s \in \mathcal{S}. \tag{20}$$

At the controller node, all network links' capacities can be determined as discussed in section II. As the network starts serving requested data flows, the remaining capacity of each link is recalculated as,

$$\overline{R_{v,v',k}} = R_{v,v',k} - \sum_s R_s \mathbb{1}_{\{\mathcal{E}_{v,v',k}\in\rho_s\}}, \tag{21}$$

where ρ_s is the selected route for the previously selected flows, and $\mathbb{1}_{\{\cdot\}}$ is an indicator that equals one if the argument in brackets is correct, and zero otherwise. At each slice admission iteration, new data flow could be admitted and previously allocated flows could be released. Hence, the remaining (i.e., unoccupied) network link's capacities are updated. The feasible flow rate R_s for each nominated route is updated based on $\overline{R_{v,v',k}}$. If the revised slice rate becomes less than the threshold SLA rate, the nominated route is eliminated. We select a route $\rho(\overline{R_s})$, with capacity occupation percentage $\overline{R_s}$ reserved at all route links. Note that the flow cannot be admitted if all nominated routes table entries are withdrawn.

Based on the candidate routes obtained in the previous subsection, we express the potential congestion for each network link based on the summation of all nominated route rates going through that link. The potential congestion per network link is expressed as,

$$Y_{v,v',k} = \sum_s \sum_{\tilde{\rho}_s\in\boldsymbol{\rho}_s} R_{v,v',k} - \sum_s R_s \mathbb{1}_{\{\mathcal{E}_{v,v',k}\in\tilde{\rho}_s\}}, \tag{22}$$

where $\tilde{\rho}_s$ is a nominated flow and $\boldsymbol{\rho_s}$ is the set of all nominated flow routes. We develop a loss function that penalizes data transmission over the links with high potential congestion to allow admission to subsequent flows. The loss function is expressed as,

$$L(\overline{R_s}) = \sum_{\mathcal{E}\in\boldsymbol{\mathcal{E}}} (Y_{v,v',k} - \overline{R_{v,v',k}})^+ \left(\frac{\overline{R_s}}{R_{v,v',k}}\right), \tag{23}$$

where, $(\cdot)^+ = \max\{0, \cdot\}$. We select the route that balances the achieved flow utility $U_s(\overline{R_s})$ and losses $L(\overline{R_s})$ such that,

$$\underset{\rho\in\boldsymbol{\rho}}{\text{argmax}} \quad U_s(\overline{R_s}) - w_c L(\overline{R_s}), \tag{24}$$

where w_c is a constant and $\boldsymbol{\rho}$ denotes all nominated routes. The value of w_c is selected by applying a bisection optimization algorithm that maximizes the utility function in (8). That is done by systematic trial and error procedure where (24) is solved for different w_c values and then, the solutions are used to obtain U in (8) for each value of w_c. New values of w_c are examined based on the bisection optimization algorithm until maximum utility, U is obtained. This procedure is done only one time for given network architecture. The procedure of the resource allocation sub-problem is summarized in Algorithm 2.

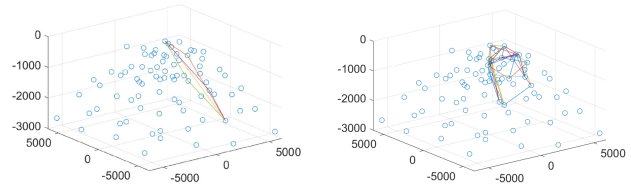
Algorithm 2 Route Selection and Resource Allocation

```
Update R̄_{v,v',k} and Ȳ_{v,v',k} as in (21) and (22)
for all flows in the flows pool do
    Randomly select a flow, s, with probability as in (20)
    for all selected flow nominated routes do
        Update R_s = min R_{v,v',k}, ∀ρ̃_s ∈ ρ
        if updated R_s < R_s^th:
            Remove route from "nominated routes" table
        end if
    end for
    if the "nominated routes" table of the flow s is empty:
        Reject the flow s
    else: Calculate L(R̄_s) as in (23)
        Select route and resource allocation by solving (24)
    end if
    Update R̄_{v,v',k} and Ȳ_{v,v',k} as in (21) and (22)
end for
```

Before discussing numerical results, we provide algorithm complexity analysis and practical implementation notes. The algorithm complexity is of order $\mathcal{O}(N^{\hat{H}})$, where N is the number of neighbors and $\hat{H}$ is the number of hops. $\hat{H}$ depends on T_{th} and a predefined maximum number of hops (H), while N depends on R_{th}. This complexity is significantly reduced by considering the elimination rules for well-defined constraints. Although the algorithm complexity is exponential, the real implementation converges quickly for a mid-size network ($\sim$100 nodes) which is suitable for underwater networks. For larger networks, multiple controllers could be used to manage network slicing. Note that the route nomination is done only at the controller (supplied with a powerful processing unit) and is done over long periods of time (can be hours or days depending on the dynamicity of the network). Further, the low data rate and long propagation delay lead to extended computational time at the controller per slice evaluation period. To guarantee the convergence of the algorithm, a minimum R_{th} and a maximum T_{th} can be defined. This algorithm allows for improved performance optimization for the strictly limited underwater network resources.

V. NUMERICAL RESULTS

In this section, we verify the network operation, obtain insightful analysis, and test our ANS solution performance for different UACS scenarios. A 3D underwater network comprised of one controller at the sea surface and 80 nodes at several depths is simulated. The underwater nodes are placed over a pyramid of 2700m height and 3500m base radius, as shown in Fig. 5. The controller is positioned at the summit of the pyramid. At low water depths, a higher node density is assumed to meet increased data rate demands by the controller. To model for underwater node drifting owing to ocean currents, each node's position is displaced by a



(a) A MANO flow nominated routes. (b) An SM flow nominated routes.

FIGURE 5. Nominated routes for MANO and SM slice types.

random offset. The solution and the simulation account for nodes mobility through an additional nodes position offset that is added over time at each slice evaluation iteration. Fig. 5 illustrates an example of nominated routes for two slice types. It shows how the ANS solution nominates routes with a low number of hops to slices that require low delay such as the MANO slice, while short-distance hops are nominated for high data rate slices such as the SM slice type. This figure also demonstrates how the nominated routes table is filled to fulfill the slice requirements.

The network and the ANS solutions are simulated using Matlab. A Poisson distributed number of flow requests with an average λ are generated during each slice evaluation period. The slice type and terminals are randomly selected from a predefined set of slice types and nodes. At each slice evaluation iteration, the ANS algorithm is performed to grant network admission and resources to a subset of flow requests that are in the flows pool. Average performance is obtained by running the simulation for a large number of slice evaluation iterations ($>$1000 iterations). We assume in the simulation that the maximum flow admittance delay is equal to one slice evaluation period, i.e., not admitted flows are directly removed from the flows pool and rejected. Unless otherwise mentioned, the default system parameters are shown in Table 4. In the table, a_{th}, b_{th}, α_1, and α_2 are the utility function parameters. λ is the average of flow requests at each slice evaluation period. m_s is the number of slice evaluation periods over which resources are given to admitted flows. The notation $\eta_{\text{e/s}}$ denotes electric to sound conversion efficiency, and P_{elec} is the electric transmission power. w_c is a constant to balance flow utility and congestion avoidance. L_A is a percentage representing remaining link availability after considering practical issues, such as modulation and coding, and MAC layer efficiency. The rest of the parameters are related to the transmission and environment characteristics. Noise variance, N, and absorption coefficient, $\alpha(f)$ are modeled as in [41] and [45], respectively.

We consider four types of slices: (1) management and network orchestration (MANO) traffic to handle control signaling, (2) emergency communication (EC) to allow for reduced delay and high priority traffic, (3) automation and communication (AC) to provide high capacity flows, and (4) sensing and monitoring (SM) slices to allow high delay tolerant traffic. The slices and their SLA are outlined in Table 5. The

TABLE 4. Default system parameters.

Parameter	Value	Parameter	Value
α_1	0.5	α_2	0.5
a_{th}	0.3	b_{th}	0.3
λ	80	m_s	5
w_c	10^{-8}	L_A	50%
P_{elec}	1 [W]	$\eta_{\text{e/s}}$	80%
tx. band k_1	$[2, 12]$ kHz	tx. band k_2	$[20, 60]$ kHz
κ	1.5	packet size, b	1000 bits

TABLE 5. SLA parameters for different slice types.

Parameter	description	Value			
	Slice type	MANO	EC	AC	SM
w_s	Slice priority	0.3	0.4	0.2	0.1
R_s^{t} [kbps]	Target slice rate	30	70	200	150
R_s^{th} [kbps]	Threshold slice rate	10	35	100	75
t_s^{t} [s]	Target slice delay	0.1	0.1	0.5	0.7
t_s^{th} [s]	Threshold slice delay	0.2	0.2	1	1.2
$w_{s,r}$	slice rate utility weight	0.2	0.2	0.5	0.8
$w_{s,t}$	slice delay utility weight	0.8	0.8	0.5	0.2

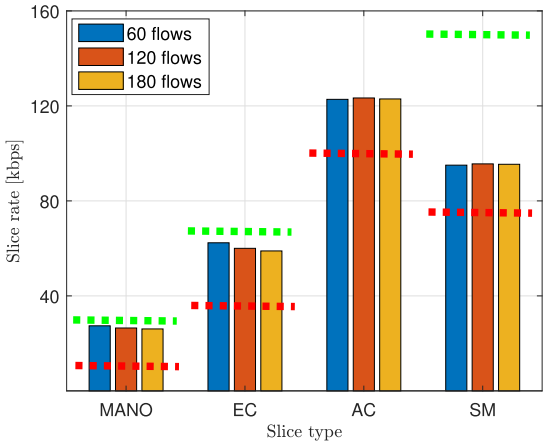
threshold and target time delays are defined as follows:

$$T_s^{\text{th}} = t_s^{\text{th}} + T_s^{\min},$$
$$T_s^{\text{t}} = t_s^{\text{t}} + T_s^{\min},$$

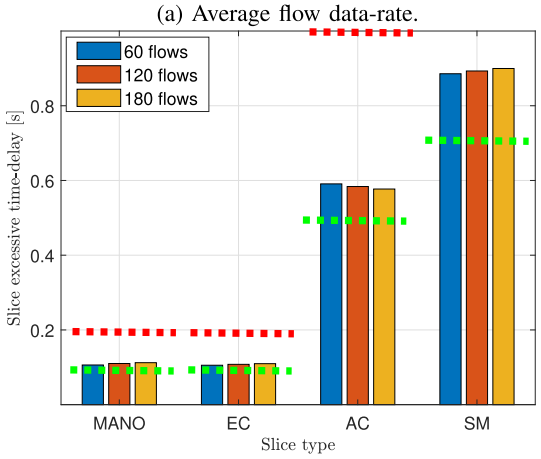
where, $T_s^{\min} = \frac{d_{v_{src},v_{dst}}}{c} + \frac{b}{R_s^{\text{th}}}$ is the minimum feasible time delay for transmitting data over the distance between the flow source and the destination, and $t_s^{\text{th}}(t_s^{\text{t}})$ are the slice SLA threshold (target) excessive delays, due to increased transmission distance over multi-hops. In the UACS, the needed delay is greatly dependent on the transmission distance due to slow propagation speed. The SLA time delay restrictions are set up in this way to minimize excessive and needless delay for short-range flows, while yet allowing long-range communications. The traffic generated by the MANO and EC slices is limited but should be conveyed as quickly as possible and with a high priority. We assume that MANO and EC traffic goes either from or to the controller. On the other hand, high transmission delays are acceptable for the AC and SM slices, but high transmission rates are necessary to send a comparatively large quantity of data. The AC flows can be required between any two nodes, but the SM flows are sent to the controller directly or indirectly (through several flows). For transmissions between underwater nodes and terrestrial/aerial nodes, communication is established through the controller, which acts as a gateway for the underwater network. The communications between the controller and terrestrial/aerial nodes are managed by other systems. The utility function parameters and the slice reservation period are fixed for all slice types, as shown in Table 4. This simplifies the presentation of system analysis and makes unbiased conclusions. The simulation can be adjusted to consider different values that might reflect a given practical scenario.

A. SOLUTION VALIDATION

In this subsection, we validate that the admitted flows are provided with the required SLA rate and time delay thresholds. We also show that the admitted slices do not consume



(a) Average flow data-rate.



(b) Average flow excessive time-delay.

FIGURE 6. **Average flow rate and excessive delay for different slices and number of flows.**

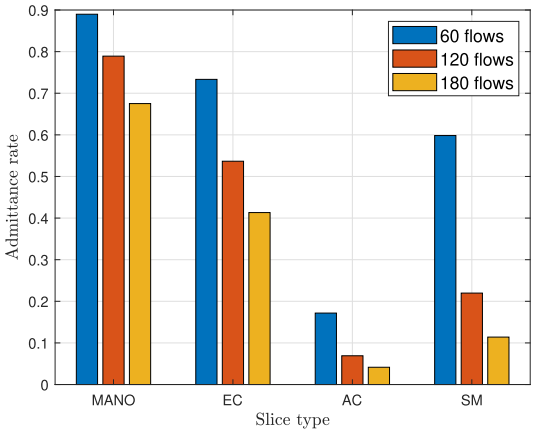
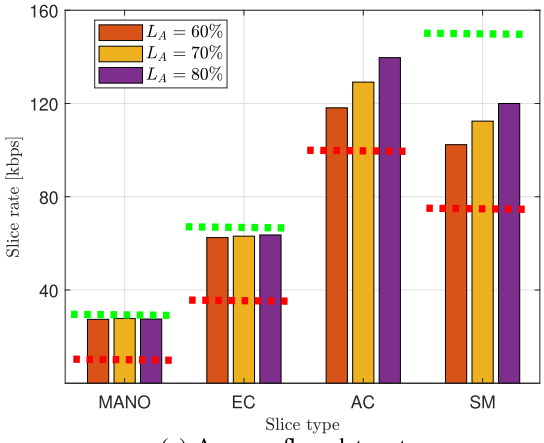
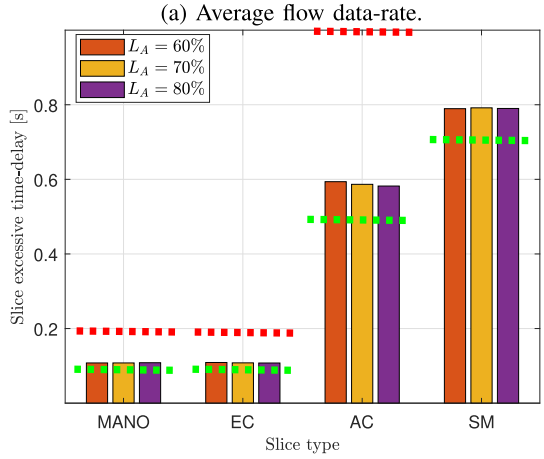


FIGURE 7. **Admittance rate vs slice type and number of flows.**



(a) Average flow data-rate.



(b) Average flow excessive time-delay.

FIGURE 8. **Average flow rate and excessive delay for different slices and link availability percentages.**

more resources than the SLA targets. In Fig. 6, the flow rates and time-delays are shown for all slice types and different number of flows. In Fig. 6 and Fig. 8, the dashed green and red lines represent the target and threshold SLA values, respectively. Due to their high SLA priorities, the MANO and EC slices are closer to the target SLA requirements than the AC and SM slices. As the number of flows increases, the rate and the excessive time-delay per flow are not changed considerably. However, the flow admittance rate dramatically decreases as the number of flow requests increases, as shown in Fig. 7. The flow data rate and time delay for each slice type is also shown for different link availability percentages, in Fig. 8. As L_A increases, the data rate of admitted flows approaches the target slice rate, especially for the high data rate slice types. However, the time delay is slightly improved. Next, we analyze the utilization of the transmission bands, k_1 and k_2. In Fig. 9, the utilization of transmission bands per admitted flows is shown against the average number of flows. The curves show the number of admitted flows that are transmitted over routes that include links from the band k_1 only, band k_2 only, and both bands. Since most admitted flows are MANO and EC, a high number of flows are transmitted over the band k_1 which allows long-distance communication and reduced latency. As the required data rate increases, more flows are admitted by allocating resources from the band k_2. Interestingly, the route of some admitted flows includes links from both transmission bands, to balance achieved performance against SLA requirements and to avoid traffic congestion. Moreover, notice how the number of admitted flows saturates with the increase of the number of

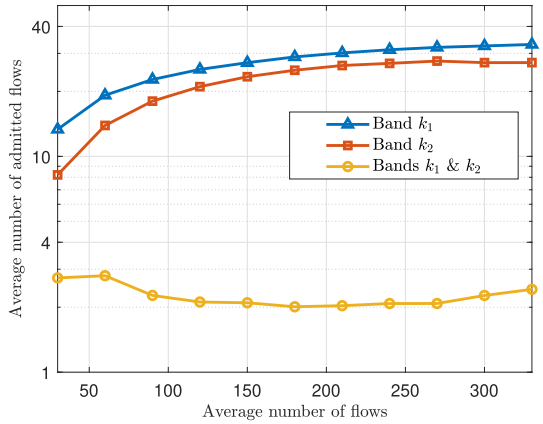


FIGURE 9. Utilization of transmission bands per admitted flows.

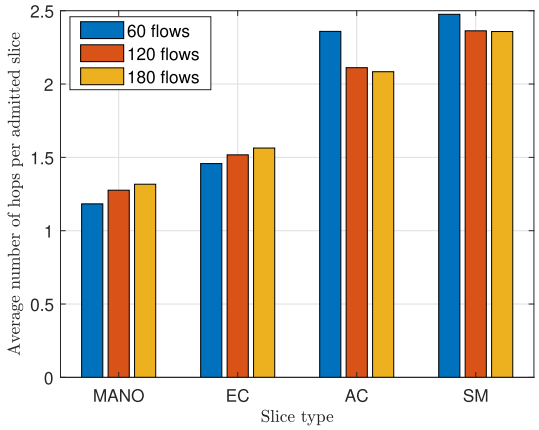


FIGURE 11. Average number of hops per slice type for different number of flows.

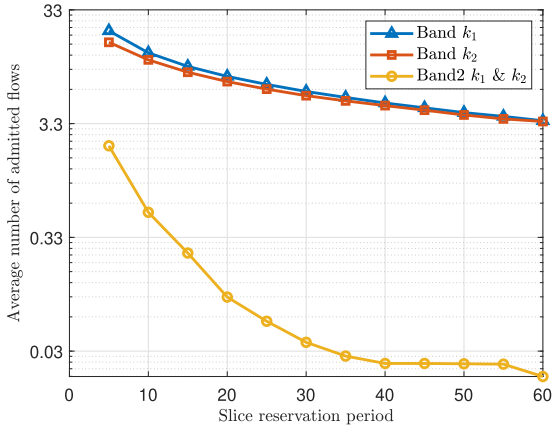


FIGURE 10. Utilization of transmission bands per admitted flows vs slice reservation period.

required flows, as the network resources get highly utilized. Fig. 9 illustrates how the different channel features support the slices' requirements. Since low acoustic frequency signals can propagate to higher distances, a high number of flows are transmitted over the band k_1. On the other hand, slices that require a high data rate are expected to be transmitted over the k_2 band. The average number of admitted flows over transmission bands is also shown against the slice reservation period in Fig. 10. As the reservation period increases per admitted flow, a smaller number of flows can be admitted. Notice how in 9 and 10, the number of flows transmitted over both bands, k_1 and k_2, is low. The complexity of the ANS algorithm can be reduced by filling the nominated routes tables for each band separately, and therefore, eliminating the possibility of a flow transmission over both bands. Because the number of flows transmitted over both bands is low, the performance degradation of eliminating this possibility might be acceptable. In Fig. 11, the average number of hops is shown against slice types. As expected, the number of hops tends to be low for MANO and EM slice types to minimize the transmission delay. On the other hand, AC and SM slice types have a higher number of slices as higher data rates are required for these slice types.

B. ANS PERFORMANCE EVALUATION

In order to evaluate the proposed ANS solution, we compare its utility performance with a next shortest path (NSP) algorithm designed for a terrestrial vanilla-SDN network [42]. Depending on the link's remaining capacity, the NSP algorithm selects the path that provides maximum utility. However, potential traffic by unassigned flows, slice priority, and balancing of slice time-delay vs slice rate are not provided in the NSP algorithm. Since there are no special SDN algorithms for underwater communication, up to the authors' knowledge, we compare our algorithm with a vanilla-SDN algorithm that is designed for terrestrial networks. The PHY layer is modified to guarantee a fair comparison is conducted. Note that comparison with terrestrial NS is not possible since terrestrial NS considers a more complicated system architecture, i.e., over core and radio access network and with distributed network functions. The normalized utility of the ANS and NSP algorithms is shown versus the average number of network flows per slice evaluation period in Fig. 12. The overall system utility rises as more flows are required since the remaining network resources are verified against more flows that might utilize unoccupied links. As the number of flows increases, the percentage of performance improvement realized by employing the ANS algorithm over the NSP algorithm grows, reaching a maximum of 15% when the number of flows is 210 flows per slice evaluation period. This is due to the increased necessity to consider other traffic flows when network congestion develops. The performance boost begins to dwindle when the network resource becomes saturated due to the growing number of flows. We notice that the ANS algorithm always achieves higher utilities than the NSP algorithm with maximum gain percentages when the network resources are moderately utilized. Fig. 13 compares the number of admitted flows for our ANS solution against the NSP solution. The number of admitted flows is up to

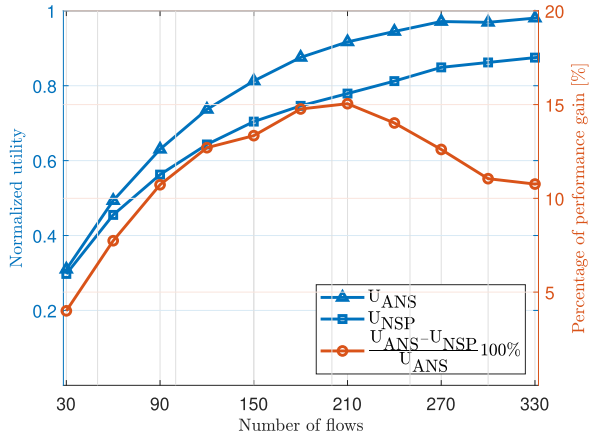


FIGURE 12. Comparison between ANS and NSP networks' utilities against the average number of flow requests.

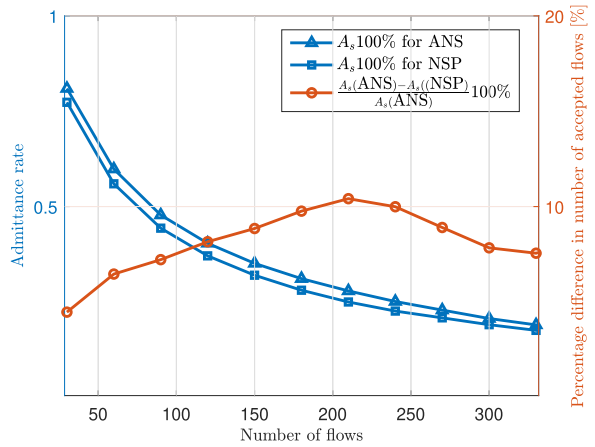


FIGURE 13. Comparison between ANS and NSP networks' flow admittance rate against the average number of flows.

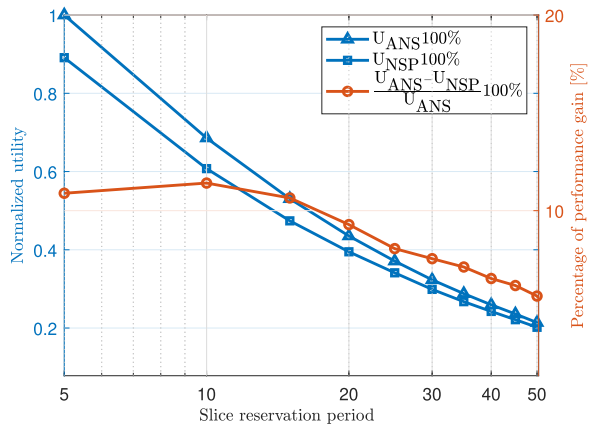


FIGURE 14. Comparison between ANS and NSP networks' utilities against slice reservation period.

10% higher when the ANS solution is used. The admittance curve follows a similar trend as the utility gain curve shown in Fig. 12.

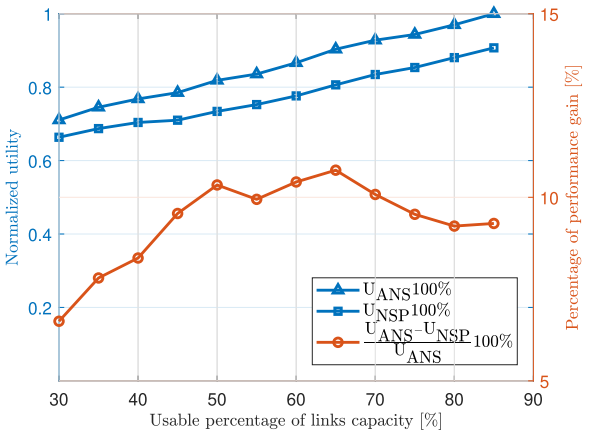


FIGURE 15. Comparison between ANS and NSP networks' utilities against link bandwidth availability percentage.

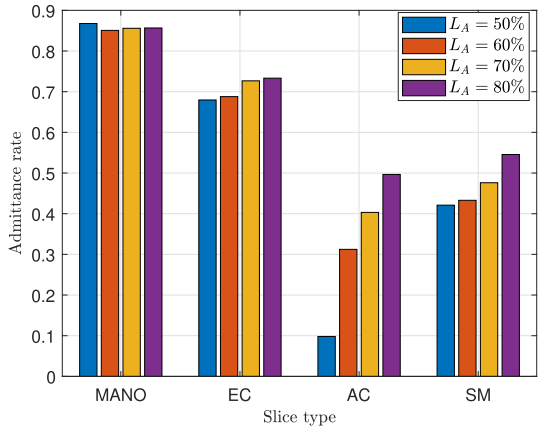


FIGURE 16. Admittance rate for different slice types and link bandwidth availability.

Similarly, the utility gains of the ANS and the NSP algorithms against the slice reservation period and link availability are shown in Fig. 14 and 15, respectively. As the slice reservation period increases and/or the network links availability percentage decreases, the network resources become more scarce. Therefore, the utility gain of applying the ANS over NSP is as low as 5% when the slice reservation period is $50T_0$ and around 6% when $L_A = 0.3$. In Fig. 15, it is possible to notice that the utility gain increases as L_A increases until 50%, before it decreases and increases again between $L_A \in [50\%, 65\%]$, and finally decreases after $L_A > 70\%$. This counter-intuitive behavior is due to the utilization of two different transmission bands. To support this reasoning, we show in Fig. 16 the flow admittance rate for different slice types and link availability. When the link availability is 50%, the admittance rate of high-rate slices is as low as 0.1 for the AC slice type, and 0.4 for the SM slice type. At this link availability, the resources of the band k_2 are scarce relative to the admission demand. On the other hand, the admittance rate for the MANO and EC slice types are as high as 0.85 and 0.7 respectively, which implies the abundance

of band k_1 resources to carry low latency slices. As the link availability increases from 50% to 70%, less gain is achieved in the k_1 band and more gain in the k_2 band. Finally, as the link availability increases beyond 70%, the gain of utilizing ANS over NSP saturates to around 9%.

VI. CONCLUSION

Automatic network slicing offers globally efficient network resource management solutions, improves QoS, improves network operations, and lowers implementation costs. In this paper, a novel optimization framework for dynamic admission control, network routing, and resource allocation is developed and solved automatically based on SLAs. SLA-based slicing can assist in abstracting the complexities of slice construction and customization in order to suit various use cases and slice priorities. The problem is decoupled into two parts, using the proposed automatic slicing heuristic method. The routes that match the minimum SLA requirements are first nominated. Then we decide if a flow should be allowed or rejected, choose one of the nominated routes, and allocate network resources to optimize the utility function.

The proposed solution is tailored specifically for UACS as it assumes simple and small network architecture, depicts UAC channel characteristics, and quickly eliminates routes with high time delay. The number of nodes we are considering in this paper is realistic nowadays. For future underwater applications, an increasing number of nodes is expected.

Numerical analysis shows that indeed the proposed ANS algorithm guarantees the required performance specified by the SLA thresholds to admitted flows. In comparison to vanilla-SDN based on NSP algorithms, the proposed ANS algorithm achieves 15% better performance. While the ANS algorithm always obtains a higher gain than the NSP algorithm, maximum gain is observed when the network resources are moderately utilized. In the future, we plan to extend this work to the case of multiple controllers and test further heuristic algorithms to improve the performance gain of the system.

ACKNOWLEDGMENT

An earlier version of this paper was presented at the Fourth International Balkan Conference on Communications and Networking [DOI: 10.1109/BalkanCom53780.2021.9593185]. A U.S. patent has been filed recently [2].

REFERENCES

[1] O. M. Bushnaq, I. Zhilin, G. D. Masi, E. Natalizio, and I. Akyildiz, "Automatic network slicing for resource allocation in underwater acoustic communication systems," in *Proc. Int. Balkan Conf. Commun. Netw. (BalkanCom)*, Sep. 2021, pp. 90–95.

[2] O. M. Bushnaq, I. V. Zhilin, G. D. Masi, E. Natalizio, I. F. Akyildiz, and N. Aaraj, "Systems, methods, and computer-readable media of automatic network slicing of underwater acoustic communication system resources," U.S. Patent 17 806 283, Jun. 5, 2022.

[3] I. F. Akyildiz, P. Wang, and Z. Sun, "Realizing underwater communication through magnetic induction," *IEEE Commun. Mag.*, vol. 53, no. 11, pp. 42–48, Nov. 2015.

[4] Y. Li, S. Wang, C. Jin, Y. Zhang, and T. Jiang, "A survey of underwater magnetic induction communications: Fundamental issues, recent advances, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2466–2487, 3rd Quart., 2019.

[5] H. Kaushal and G. Kaddoum, "Underwater optical wireless communication," *IEEE Access*, vol. 4, pp. 1518–1547, 2016.

[6] Z. Zeng, S. Fu, H. Zhang, Y. Dong, and J. Cheng, "A survey of underwater optical wireless communications," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 204–238, 1st Quart., 2017.

[7] A. Al-Kinani, C.-X. Wang, L. Zhou, and W. Zhang, "Optical wireless communication channel measurements and models," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 1939–1962, 3rd Quart. 2018.

[8] N. Saeed, A. Celik, T. Y. Al-Naffouri, and M.-S. Alouini, "Underwater optical wireless communications, networking, and localization: A survey," *Ad Hoc Netw.*, vol. 94, Nov. 2019, Art. no. 101935.

[9] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Commun. Mag.*, vol. 47, no. 1, pp. 84–89, Jan. 2009.

[10] M. Stojanovic, "Recent advances in high-speed underwater acoustic communications," *IEEE J. Ocean. Eng.*, vol. 21, no. 2, pp. 125–136, Apr. 1996.

[11] G. Qiao, Z. Babar, L. Ma, S. Liu, and J. Wu, "MIMO-OFDM underwater acoustic communication systems—A review," *Phys. Commun.*, vol. 23, pp. 56–64, Jun. 2017.

[12] V. Sharma and S. Kumar, "Recent developments in MIMO channel estimation techniques," in *Proc. 2nd Int. Conf. Digit. Inf. Commun. Technol. Appl. (DICTAP)*, May 2012, pp. 1–6.

[13] B. Li, J. Huang, S. Zhou, K. Ball, M. Stojanovic, L. Freitag, and P. Willett, "MIMO-OFDM for high-rate underwater acoustic communications," *IEEE J. Ocean. Eng.*, vol. 34, no. 4, pp. 634–644, 2009.

[14] M. V. Volkov, V. A. Grigor'ev, I. V. Zhilin, A. A. Lunkov, V. G. Petnikov, and A. V. Shatravin, "An arctic-type shallow-water acoustic waveguide as an information transmission channel for underwater communications," *Acoust. Phys.*, vol. 64, no. 6, pp. 692–697, Nov. 2018.

[15] M. V. Volkov, A. A. Lunkov, V. G. Petnikov, and A. V. Shatravin, "Underwater acoustic communications using vertical receiver arrays in ice-covered shallow-water areas," *Oceanology*, vol. 61, no. 4, pp. 569–580, Jul. 2021.

[16] I. F. Akyildiz, D. Pompili, and T. Melodia, "Challenges for efficient communication in underwater acoustic sensor networks," *ACM SIGBED Rev.*, vol. 1, no. 2, pp. 3–8, Jul. 2004.

[17] S. Fattah, A. Gani, I. Ahmedy, M. Y. I. Idris, and I. A. T. Hashem, "A survey on underwater wireless sensor networks: Requirements, taxonomy, recent advances, and open research challenges," *Sensors*, vol. 20, no. 18, p. 5393, Sep. 2020.

[18] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: Research challenges," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 257–279, Mar. 2005.

[19] L. Liu, S. Zhou, and J.-H. Cui, "Prospects and problems of wireless communication for underwater sensor networks," *Wiley Wireless Commun. Mobile Comput.*, vol. 8, no. 8, pp. 977–994, Oct. 2008.

[20] A. Petroni, H.-L. Ko, T. Im, Y.-H. Cho, R. Cusani, G. Scarano, and M. Biagi, "Hybrid space-frequency access for underwater acoustic networks," *IEEE Access*, vol. 10, pp. 23885–23901, 2022.

[21] Z. Liu, X. Meng, Y. Yuan, Y. Yang, and K. Y. Chan, "Joint optimization for throughput maximization in underwater acoustic networks with energy harvesting," *Peer Peer Netw. Appl.*, vol. 14, no. 4, pp. 2115–2126, Jul. 2021.

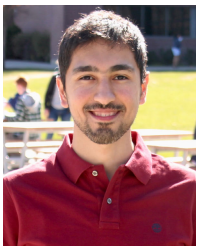
[22] M. Y. I. Zia, J. Poncela, and P. Otero, "State-of-the-art underwater acoustic communication modems: Classifications, analyses and design challenges," *Wireless Pers. Commun.*, vol. 116, no. 2, pp. 1325–1360, Jan. 2021.

[23] I. V. Zhilin, O. M. Bushnaq, G. De Masi, E. Natalizio, and I. F. Akyildiz, "A universal multimode (acoustic, magnetic induction, optical, RF) software defined radio architecture for underwater communication," in *Proc. 15th Int. Conf. Underwater Netw. Syst.*, Nov. 2021, pp. 1–6.

[24] I. F. Akyildiz, P. Wang, and S.-C. Lin, "SoftWater: Software-defined networking for next-generation underwater communication systems," *Ad Hoc Netw.*, vol. 46, pp. 1–11, Aug. 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1570870516300579

[25] E. Demirors, G. Sklivanitis, T. Melodia, S. N. Batalama, and D. A. Pados, "Software-defined underwater acoustic networks: Toward a high-rate real-time reconfigurable modem," *IEEE Commun. Mag.*, vol. 53, no. 11, pp. 64–71, Nov. 2015.

[26] R. Su, D. Zhang, R. Venkatesan, Z. Gong, C. Li, F. Ding, F. Jiang, and Z. Zhu, "Resource allocation for network slicing in 5G telecommunication networks: A survey of principles and models," *IEEE Netw.*, vol. 33, no. 6, pp. 172–179, Nov./Dec. 2019.
[27] Z. Jin, Q. Zhao, and Y. Luo, "Routing void prediction and repairing in AUV-assisted underwater acoustic sensor networks," *IEEE Access*, vol. 8, pp. 54200–54212, 2020.
[28] H. Tran-Dang and D. Kim, "Channel-aware energy-efficient two-hop cooperative routing protocol for underwater acoustic sensor networks," *IEEE Access*, vol. 7, pp. 63181–63194, 2019.
[29] Y. Chen, Y. Tang, X. Fang, L. Wan, Y. Tao, and X. Xu, "PB-ACR: Node payload balanced ant colony optimal cooperative routing for multi-hop underwater acoustic sensor networks," *IEEE Access*, vol. 9, pp. 57165–57178, 2021.
[30] *Network Slicing Use Case Requirements*, GSM Association, GSMA, London, U.K., 2018.
[31] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck, "Network slicing and softwarization: A survey on principles, enabling technologies, and solutions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2429–2453, 3rd Quart., 2018.
[32] S. Vassilaras, L. Gkatzikis, N. Liakopoulos, I. N. Stiakogiannakis, M. Qi, L. Shi, L. Liu, M. Debbah, and G. S. Paschos, "The algorithmic aspects of network slicing," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 112–119, Aug. 2017.
[33] S. Zhang, "An overview of network slicing for 5G," *IEEE Wireless Commun.*, vol. 26, no. 3, pp. 111–117, Jun. 2019.
[34] S. Wijethilaka and M. Liyanage, "Survey on network slicing for Internet of Things realization in 5G networks," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 957–994, 2nd Quart., 2021.
[35] P. L. Vo, M. N. H. Nguyen, T. A. Le, and N. H. Tran, "Slicing the edge: Resource allocation for RAN network slicing," *IEEE Wireless Commun. Lett.*, vol. 7, no. 6, pp. 970–973, Dec. 2018.
[36] Y. L. Lee, J. Loo, T. C. Chuah, and L.-C. Wang, "Dynamic network slicing for multitenant heterogeneous cloud radio access networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2146–2161, Apr. 2018.
[37] W. Guan, X. Wen, L. Wang, Z. Lu, and Y. Shen, "A service-oriented deployment policy of end-to-end network slicing based on complex network theory," *IEEE Access*, vol. 6, pp. 19691–19701, 2018.
[38] F. Tonini, C. Natalino, M. Furdek, C. Raffaelli, and P. Monti, "Network slicing automation: Challenges and benefits," in *Proc. Int. Conf. Opt. Netw. Design Model. (ONDM)*, May 2020, pp. 1–6.
[39] F. Zhou, P. Yu, L. Feng, X. Qiu, Z. Wang, L. Meng, M. Kadoch, L. Gong, and X. Yao, "Automatic network slicing for IoT in smart city," *IEEE Wireless Commun.*, vol. 27, no. 6, pp. 108–115, Dec. 2020.
[40] A. Kak and I. F. Akyildiz, "Towards automatic network slicing for the internet of space things," *IEEE Trans. Netw. Service Manage.*, vol. 19, no. 1, pp. 392–412, Mar. 2022.
[41] M. Stojanovic, "On the relationship between capacity and distance in an underwater acoustic communication channel," in *Proc. 1st ACM Int. workshop Underwater Netw. (WUWNet)*, 2006, pp. 41–47, doi: 10.1145/1161039.1161049.
[42] B. G. Assefa and O. Ozkasap, "RESDN: A novel metric and method for energy efficient routing in software defined networks," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 2, pp. 736–749, Jun. 2020.
[43] A. Celik, N. Saeed, B. Shihada, T. Y. Al-Naffouri, and M.-S. Alouini, "A software-defined opto-acoustic network architecture for internet of underwater things," *IEEE Commun. Mag.*, vol. 58, no. 4, pp. 88–94, Apr. 2020.
[44] H. Luo, K. Wu, R. Ruby, Y. Liang, Z. Guo, and L. M. Ni, "Software-defined architectures and technologies for underwater wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2855–2888, 4th Quart., 2018.
[45] H. U. Yildiz, V. C. Gungor, and B. Tavli, "Packet size optimization for lifetime maximization in underwater acoustic sensor networks," *IEEE Trans. Ind. Informat.*, vol. 15, no. 2, pp. 719–729, Feb. 2019.
[46] J. Proakis and M. Salehi, *Digital Communications*, 5th ed. New York, NY, USA: McGraw-Hill, 2008.
[47] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Perez, "DeepCog: Cognitive network management in sliced 5G networks with deep learning," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, Apr. 2019, pp. 280–288.
[48] S.-C. Lin, "End-to-end network slicing for 5G&B wireless software-defined systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–7.



OSAMA M. BUSHNAQ (Member, IEEE) received the B.S. degree in communication engineering from Princess Sumaya University for Science and Technology, Amman, Jordan, in 2012, the M.Eng. degree in electrical engineering from the University of New Brunswick, Fredericton, NB, Canada, in 2014, and the Ph.D. degree from the King Abdullah University of Science and Technology, Thuwal, Saudi Arabia, in 2020.

He was a Visiting Researcher with the Delft University of Technology, Delft, The Netherlands, in 2016, and The University of British Columbia, Vancouver, BC, Canada, in 2019. He is currently a Senior Researcher with the Technology Innovation Institute, Abu Dhabi, United Arab Emirates. His current research interests include UAV communications, underwater communications, the IoT networks, and system optimization.

IGOR V. ZHILIN (Member, IEEE) received the B.Sc. degree in applied mathematics and physics and the M.Sc. degree in applied mathematics and electrical engineering, with excellence, from the Moscow Institute of Physics and Technology (MIPT), in 2010 and 2012, respectively, and the Ph.D. degree in theoretical fundamentals of informatics from the Kharkevich Institute for Information Transmission Problems RAS (IITP RAS), in 2015. Since 2010, he has been working with the IITP RAS; his work directions included error correction codes research, modulation and coding schemes design, and physical layer field experiments. Since May 2011, he has been a Lead Researcher with the Technology Innovation Institute (TII), Abu Dhabi, United Arab Emirates. His current research interests include underwater communications, multimode communications, physical layer technology, wireless networks, and cross-layer protocols.

GIULIA DE MASI (Senior Member, IEEE) received the master's degree (magna cum laude) from the University of L'Aquila, and the Ph.D. degree from the University of Rome La Sapienza, in collaboration with the Max Planck Institute, Dresden, and the City University of London.

She was a Postdoctoral Researcher at the Polytechnic University of Marche and a Visiting Researcher at the Hitachi Research Laboratory, Nara, Japan. In 2008, she started working in the research and development field for the marine industry (Snamprogetti Center of Excellence, Italy), joining the Department of Advanced Engineering Services and Technology Innovation Projects. She worked in several academic institutions in United Arab Emirates, before joining the Technology Innovation Institute, Abu Dhabi, United Arab Emirates, as a Principal Scientist. She is currently a Principal Scientist with the Autonomous and Robotic Research Center, Technology Innovation Institute, Abu Dhabi. She is also affiliated with Khalifa University, Abu Dhabi, within the College of Engineering. She has more than 60 peer-reviewed publications (including journals and conferences) and more than 40 reports for industry. Her research interests include collective intelligence, machine learning, deep learning, and optimization with applications to underwater multi-robot systems. She has been recently awarded as the Women in Engineering (WIE) Propel Laureate IEEE by Oceanic Engineering Society in liaison with WIE.

ENRICO NATALIZIO (Senior Member, IEEE) received the master's (magna cum laude) and Ph.D. degrees in computer engineering from the University of Calabria, Italy, in 2000 and 2005, respectively. From 2005 to 2006, he was a Visiting Researcher at the Broadband Wireless Networking (BWN) Laboratory, Georgia Tech, Atlanta, GA, USA. From 2006 to 2010, he was a Research Fellow at the Titan Laboratory, Università della Calabria, Italy. In October 2010, he joined the POPS Team, Inria Lille–Nord Europe, France, as a Postdoctoral Researcher. From 2012 to 2018, he was an Associate Professor at the Université de Technologie de Compiègne, France, and a Full Professor at the Université de Lorraine, since September 2018. He is currently the Vice President of the Autonomous Robotics Research Center, Technology Innovation Institute, United Arab Emirates, and a Full Professor with the LORIA Laboratory, Université de Lorraine, France. His research interests include UAV communications and networking, robot and sensor communications with applications in networking technologies for disaster management and infrastructure monitoring, and the IoT privacy and security. He is also an Associate Editor of *Vehicular Communications* (Elsevier) and *Computer Networks* (Elsevier).

IAN F. AKYILDIZ (Life Fellow, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical and computer engineering from the University of Erlangen–Nürnberg, Germany, in 1978, 1981, and 1984, respectively. Since 1989, he has been the Founder and the President of Truva Inc., a consulting company based in Alpharetta, GA, USA. Since June 2020, he has been an Advisory Board Member with the Technology Innovation Institute (TII), Abu Dhabi, United Arab Emirates. Since August 2020, he has been the Founder and the Editor-in-Chief of the newly established of the *International Telecommunication Union Journal on Future and Evolving Technologies* (ITU J-FET). He worked as the Ken Byers Chair Professor of telecommunications, the Past Chair of the Telecom Group at the ECE, and the Director of the Broadband Wireless Networking Laboratory, Georgia Institute of Technology, from 1985 to 2020. He had many international affiliations during his career and established research centers in Spain, South Africa, Finland, Saudi Arabia, Germany, Russia, India, and Cyprus. He has been an ACM Fellow, since 1997. In June 2022, according to Google Scholar his H-index is 133 and the total number of citations to his articles is more than 134K. His current research interests include networking 2030, metaverse, hologram and extended reality communication, 6G/7G wireless systems, terahertz communication, and underwater communication. He received numerous awards from IEEE, ACM, and other professional organizations, including Humboldt Award from Germany and Tubitak Award from Turkey.