applied sciences

Article

An Integrated Cognitive Radio Network for Coastal Smart Cities

Huma Ghafoor [1] and Insoo Koo [2,*]

[1] School of Electrical Engineering and Computer Science, National University of Sciences and Technology (NUST), Islamabad 44000, Pakistan
[2] School of Electrical Engineering, University of Ulsan (UOU), Muger-Dong San-29, Ulsan 680-749, Korea
* Correspondence: iskoo@ulsan.ac.kr

Received: 5 July 2019; Accepted: 23 August 2019; Published: 30 August 2019

Abstract: The integration of different networks has attracted significant attention in academia. Both terrestrial and maritime communications systems have been attracting keen interest for ways to deal with various applications. As the environment of cognitive vehicular and maritime networks is extremely dynamic, these networks suffer with a long delay because of intermittent links while providing services for different applications. To this end, here we introduce the integration of cognitive vehicular and maritime networks to design a coastal smart city by utilizing software-defined networking, network function virtualization, and fog computing under the same infrastructure. This novel integrated cognitive coastal city fulfills the demand of each application user in a hybrid environment with a quicker response time. The idea is to combine vehicular and maritime communications to meet different user demands. Different virtual networks are launched by network function virtualization, and are managed and controlled by a software-defined networking controller. From the integration of software-defined networking, network function virtualization, and fog computing, both vehicular and marine users are provided with stable paths to meet each application's demands.

Keywords: cognitive radio; maritime communications; vehicular communications

1. Introduction

A coastal smart city is an up-and-coming architecture that allows for hybrid communications with a view to enable imaginative and innovative services for onboard users. A coastal city involves both terrestrial (vehicular) and maritime communications, thereby serving terrestrial and marine users by providing efficient and reliable paths. With the aim of enabling several vehicular and maritime applications (e.g., road safety and shipment to entertainment-associated information), and to fulfill all the expectations in providing services from each of these applications, here we introduce an integrated cognitive architecture. We citizens are actually exposed to a plethora of mobile applications for vehicular and maritime networks. This increase in the number of smart devices has caused difficulty in maintaining stable networking and in meeting the increasing demands of different users. Centralized cloud computing has been used as a straightforward alternative to implementing complicated services; however, due to latency constraints in different networks, cloud computing is not a good remedy for these issues [1]. Fog computing (FC) is a technology that brings services nearer to the end user, thereby improving end-to-end latency. Because the nature of both vehicles and ships in these networks is extremely dynamic, another issue is establishing a stable route to provide diversified services for different network applications. Software-defined networking (SDN) and network function virtualization (NFV), two emerging notions that are gaining popularity in both academia and industry, have inspired a good way to solve this problem [2]. Moreover, dedicated spectrums for automobile and maritime communication systems have been found to be insufficient due to an inevitable increase in the number of both vehicular and marine users and the demands of each of these applications [3,4]. The cognitive radio seems to be a corrective approach to overcome spectrum shortage issues in different networks. Consequently, to maintain stable networking and meet the increasing demands of mobile data traffic, network engineers are required to provide an effective solution to improve users' experiences. This means that citizens need an all-in-one framework to meet their growing needs in a more effective and efficient way.

To cope with the inevitable increase in data traffic, integration of SDN and NFV seems to be a viable solution to launching and managing virtual networks (VNs) on demand and at greater speed, respectively. NFV is basically a transition from proprietary hardware-based solutions to multi-vendor open solutions. NFV is like a soft appliance that can be installed on demand, whereas SDN is smart plumbing that can be changed on command. The integration reduces provisioning time from months to minutes, reduces costs, improves service-request response times, reacts faster to changing services (allowing right-sized deployments to customers), and offers competitive services. Due to this integration, the changing needs of virtual network functions (VNFs) are now automatically followed by network configurations, and an authorized person is able to track the owner of a service and the reasons why these certain configurations are required. A VNF is the virtualization of a certain NF that should operate independently. NFs are actually services that are deployed by NFV. SDN is a critical component in the majority of NFV deployments. NFV offers silo-free services (i.e., a common-layer platform with compute, storage, and network resources) where there is no duplication for each service. The infrastructure supports NFs across many geographic locations. Both NFV and SDN are complementary technologies that simplify network management and that can be applied to different network types. NFV can use SDN as a part of service function chaining (SFC), and SDN can provide connectivity between VNFs [5,6].

Previously, in order to add new features in the network, network engineers needed to purchase new equipment from different vendors. NFV solved this problem by providing network services in virtual machines, each performing different operations. Hence, NFV increases the speed of network expansion and network flexibility for the fast delivery of services. A new framework called HomeCloud [7] was proposed in order to focus on open and efficient application delivery to the edge cloud by integrating SDN and NFV. Similarly, Ying et al. [8] proposed an integrated framework that can enable dynamic orchestration of network resources to improve the performance of applications for smart cities. A regional cooperative fog computing-based intelligent vehicular network (CFC-IoV) architecture [9] was proposed for dealing with big data in the internet of vehicles (IoV) in a smart city. Different services for IoV applications were discussed in this architecture, with different resource management hierarchical models. Yang et al. [10] considered FC as an ideal complement to, rather than a substitute for, cloud computing by proposing an SDN-enabled framework for cloud-fog interoperation. The heterogeneous network (HetNet) [11] was proposed in order to integrate heterogeneous satellite-terrestrial networks to improve network elasticity by using SDN and NFV.

Collaborative vehicular edge computing (CVEC) [12] is a framework that enables edge computing to collaborate for efficient vehicular networks. It considers multiple edge computing solutions, one of them being fog computing. In the paper cited, the authors proposed the CVEC framework and discussed its architecture, principles, and special cases. Moreover, the authors used NFV and SDN as a potential technical enabler to support the CVEC framework in reality. Similarly, an integrated architecture for software defined and virtualized radio access networks with fog computing was proposed in [13] to enable network-level virtualization. Two hierarchical control levels were adopted to integrate SDN and network virtualization with fog computing. None of these architectures consider cognitive routing as one of the services provided by the NFV infrastructure in a combination of vehicular and maritime networks in a coastal city.

Here, we integrated SDN, NFV, and FC to reduce complexities in the existing infrastructure. These complexities have high (unpredictable) latency and put high pressure on the bandwidth due to explosive increases in data traffic, intermittent connectivity in vehicular and marine networks, and inconveniences in applications (that is, no exchange of information among applications). Because an NFV infrastructure supports NFs (network services deployed as virtual functions) across many geographic locations, we therefore took advantage of this notion and implemented it in a combination of vehicular and maritime networks in a coastal city. In this case, two types of applications dynamically received network resources due to the integration. Basically, the purpose of the integration was to route traffic between service functions. NFV provides network services in virtual machines working in cloud infrastructures [6]; therefore, for this proposed architecture, we used FC to preserve latency. In this paper, we present our vision for the design of an integrated coastal smart city. A user is able to operate different applications in parallel, such that a service running in one VN cannot interfere with another [14]. We will show how SDN, NFV, and FC can be integrated in a coastal city.

In the following, we consider two different network applications for coastal smart cities. Each VN represents a network. A common network with different environments is virtualized by a hypervisor into several VNs to meet the demands of each application service. The hypervisor is a virtualization controller that allocates and manages the resources (forwarding tables in this paper) over the fog cloud (a local cloud providing resources near the edge users), and launches VNFs dynamically. This means that a large coastal city network is divided into two smaller independent networks. The service provided by these VNs is routing, which means the two environments can establish the same routing path for different vehicular and marine applications. We will extend the applications for both vehicular and maritime networks, along with different services in the near future. Our motive is to apply the idea of integration in a cognitive coastal city that combines vehicular and maritime networks to provide a stable path between the source car and destination ship for hybrid communications. This hybrid communication could be required for cargo transport, establishing data links between different types of transportation and security control, as well as for surveillance. To the best of our knowledge, this is the first study that has considered a routing service in hybrid communications for a cognitive coastal city with the integration of SDN, NFV, and FC.

The main contributions are as follows:

- An integrated coastal smart city is designed in which a user is able to fulfill the service demands from different applications in less response time. This concept of integration also enables different users from different applications to be served at the same time.
- We provide hybrid, as well as pure communications between the vehicular and maritime networks by using an integration of SDN, NFV, and FC.
- To alleviate the issues of spectrum scarcity for both automobile and maritime communications, we also consider the concept of cognitive technology to maintain stable networking.

The categorization of the remaining paper is as follows. An integrated cognitive hybrid communications framework for a coastal city is presented in Section 2. In Section 3, we discuss the simulation results of the proposed coastal smart city, while Section 4 provides the conclusion of the paper.

2. Integrated Cognitive Hybrid Communications for a Coastal City

An integrated cognitive radio coastal city which can fulfill the service demands of citizens in less response time is shown in Figure 1. The objective of this proposed architecture is to reckon with key issues in different types of current applications that consider different communication systems. We designed an integrated coastal city to serve citizens with different application demands within an all-in-one framework. For the current work, we combined terrestrial and marine applications to make an integrated coastal city. The terrestrial network includes vehicular applications. The fog cloud functions as the local cloud by serving as a central entity near the edge users to provide stable links between users on ships and in vehicles. Two types of application services were launched by the NFV hypervisor: vehicular applications and maritime applications. The launched virtual functions were monitored and controlled by an SDN controller. The SDN controller is also responsible for connectivity among these launched virtual functions. It keeps the global updated network topology of both networks by communicating with the local controllers (LCs) of each network. The LCs, which are roadside units (RSUs) and ships near the shore, keep the local updated topology of the corresponding network. For marine communications, we assumed a ship moving close to shore was an LC. A border gateway (BG) was used to communicate between the two different networks. Like the LCs, the BG is in direct communication with the SDN controller.

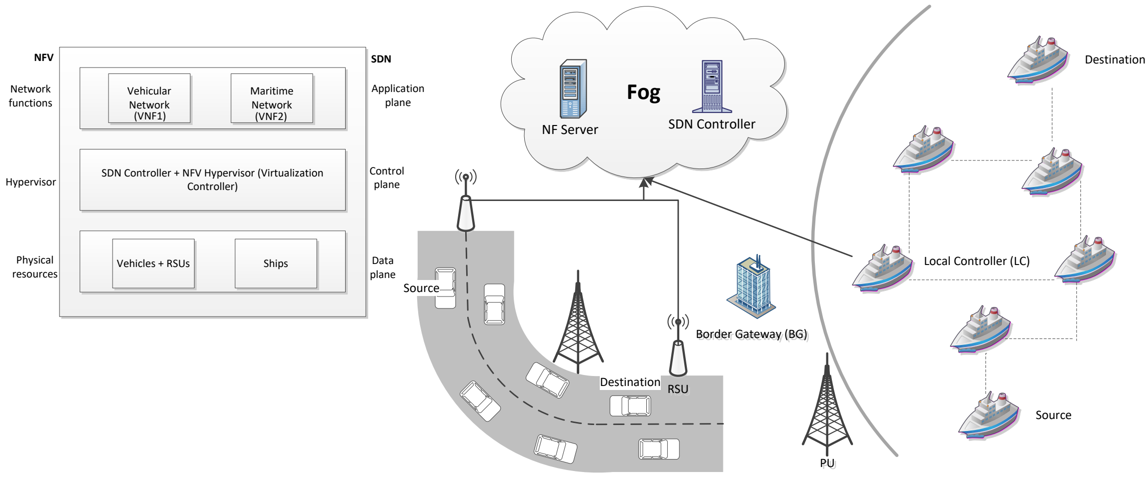


Figure 1. Integrated cognitive coastal city.

Our objective was to provide hybrid, as well as pure, communications between the vehicular and maritime networks by using an integration of SDN, NFV, and FC. For hybrid vehicle-to-ship (V2S) communications, the BG is used to provide connectivity between the two different environments, and the SDN controller is responsible for maintaining connectivity among the different VNFs. However, for vehicle-to-vehicle (V2V) or ship-to-ship (S2S) communications, the corresponding VNF provides the service to meet the users' demands. With either hybrid communications or pure communications, the fog cloud acts as a local cloud to provide flexibility and preserve latency. Accordingly, whenever an LC receives a service request message (SRM), it first examines its forwarding table to determine whether it can provide a path that fulfills the user's demand. If so, it sends the service provision message (SPM). Otherwise, it forwards the request to the SDN controller which, if it does not find the required service in its forwarding table, communicates with the VNFs to meet the user's demand. If the SDN controller does not find the respective VNF, it sends a request to the NFV hypervisor to launch the respective VNF. Figure 2 shows the flowchart for different communications with integrated technologies.

As this is a cognitive coastal city, we assumed that primary users were located across the lanes and the seashore, as shown in Figure 1. For that reason, a car can only communicate with the LC and with its neighboring cars if there is at least one common free channel between the two communicating nodes. Similarly, ships only communicate with their LC and with each other if there is a common channel free from PU activity between the two. The integration enables different users from different applications to be served at the same time. In the following, we explain in detail how a source node (either vehicle or ship) selects the best path by discussing the pure, as well as hybrid, communications using our example integrated cognitive coastal city scenario.

In Figure 1, a vehicular source-destination pair is looking for a stable route to communicate with each other. The source car asks the LC to provide a stable route. The LC provides the route either from its forwarding table or after receiving it from the SDN controller. At the same time, a source ship at sea requests a route for a destination. In a similar manner, the marine LC provides the route, either from its forwarding table or after receiving it from the SDN controller. The cars and the ships in these two networks periodically update their neighboring nodes or the LC (if within transmission range) to update each other with the current network state. Because of this exchange of data, all LCs in both networks are able to keep a local updated topology about the corresponding network.

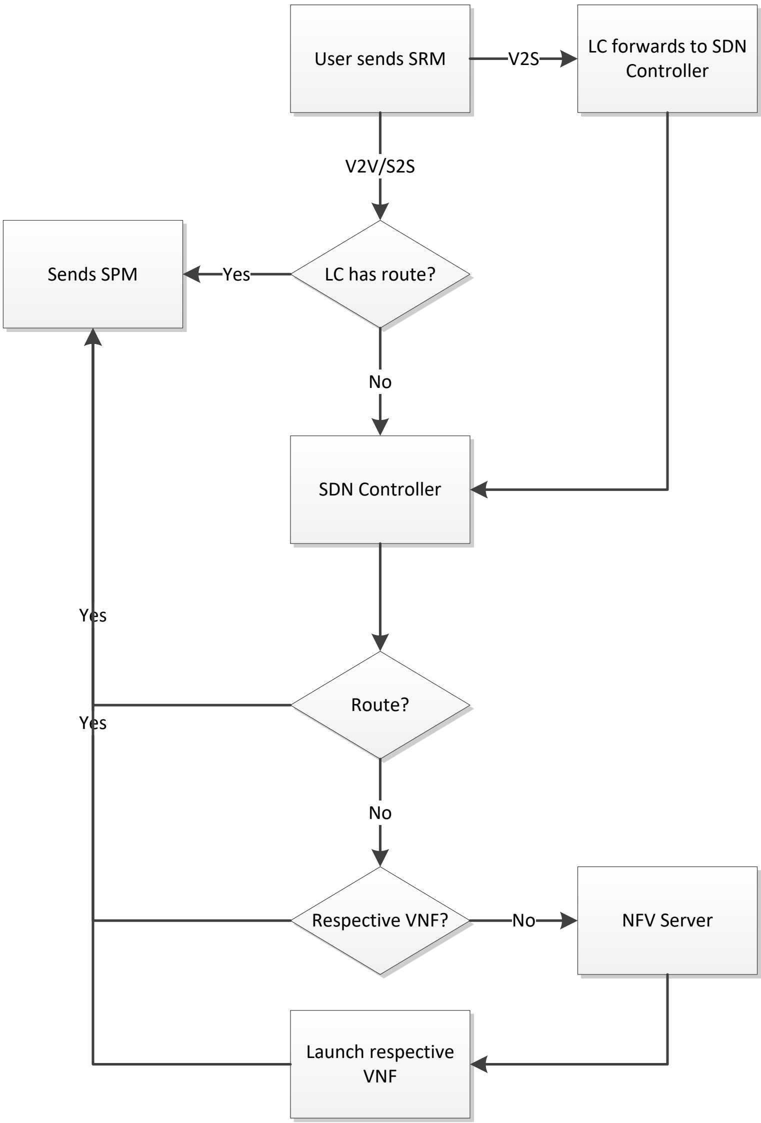


Figure 2. Flowchart representing vehicle-to-vehicle (V2V), ship-to-ship (S2S), and vehicle-to-ship (V2S) communications with integrated technologies.

Figure 1 also represents a special scenario where a car's driver wants to know about the current status of a shipment on a cargo ship (i.e., the same destination ship in Figure 1). The car needs a stable path to the destination cargo ship, and therefore, it communicates with the nearest RSU requesting a stable path to the cargo ship. Because the destination belongs to the marine network, the RSU directly forwards the request to the SDN controller. The SDN checks its forwarding table to determine whether it can fulfill the user's demand. If so, it quickly sends a service provision message to the car. Otherwise, it obtains the service from the VNF of the marine network and provides a route for hybrid communications between the source car and the destination ship. As this is a hybrid communications route, the BG is used as a gateway between the vehicular and marine environments. Hence, the final route is:

source car → relay car → destination car → BG → LC → relay ship → destination ship

One might ask why the source car did not opt to have the nearest LC for relaying the packet to the BG more quickly. Yes, it can do so more rapidly, but the primary aim of the LCs in the proposed architecture is to maintain the local updated topology to directly communicate with the SDN controller. It only serves as a gateway between the querying node and the SDN controller. In rare cases, when the network is sparse, it can serve as a relay node. This is for the reason that we do not put the LCs out with information other than that needed for their role as a local controller, and also by reason of the cost, where we do not place large number of LCs in the network.

The path is selected using a Q-learning algorithm, which is a reinforcement-learning algorithm which can be used to solve the routing problems [15]. Q-learning is based on the value of a state-action pair $Q(s,a)$, which is an updating function given as:

$$Q(s,a) = R + \gamma \sum_{s' \in S} P_{ss'} maxQ(s',a') \tag{1}$$

where γ is the discount factor, $maxQ(s',a')$ models the maximum expected future reward, and R is the expected immediate reward. Hence, the inquiring node opts the next relay node based on the following reward equation:

$$R = \beta \times (1 - \frac{1}{hello\_messages/t}) + (1-\beta)\frac{1}{TD} \tag{2}$$

where β is a weight factor, and as in our previous work [16], transmission delay, TD, is defined as:

$$TD = \frac{r \pm d_{ij}}{\Delta v} \times min(Ch_1, Ch_2, ..., Ch_M) \tag{3}$$

$\Delta v = \sqrt{(v_i \cos\theta_i - v_j \cos\theta_j)^2 + (v_i \sin\theta_i - v_j \sin\theta_j)^2}$, where v represents velocity, θ is the angle, d_{ij} is the distance between two nodes, and r is the transmission range of nodes; $min(Ch_1, Ch_2, ..., Ch_M)$ signifies the highest belief channel among all the idle channels between node i and node j. Since the path between source and destination is a cognitive routing path, two communicating nodes can therefore only make a link with each other if the two have at least one common idle channel. In view of that, spectrum sensing is essential for these nodes. The following subsection explains how vehicles and ships perform spectrum sensing and how both terrestrial and marine LCs maintain the information about the channel state for both vehicular and maritime networks.

Belief Propagation-Based Channel Selection Algorithm

As in our previous works [16,17], we performed spectrum sensing in this integrated cognitive architecture to detect the presence of PU when vehicles are moving on roads and ships are moving either near the shore or in deep sea. The subsection describes in detail the way in which vehicles and ships find the idle spectrum by sensing the environment, and how they combine their individual sensing results with each other and with their corresponding LC. A sensing table is generated by each vehicle and ship when it stores the sensing results obtained by periodically sensing the spectrum. Belief propagation [18] is a message-exchanging algorithm that aggregates beliefs by exchanging individual messages iteratively. The algorithm first computes the local sensing results for each node in the network. This is done by spectrum sensing where each node (vehicle and ship) uses an energy detection scheme to detect the existence of the PU. PUs are incumbent users utilizing licensed bands, such as high-frequency and very high-frequency, and can be any automobile traveler or other consumers located in places such as houses, shops, or restaurants for the terrestrial environment [19] and things such as oil/gas platforms or large vessels for the marine environment [20]. The TV spectrum was considered as the cognitive radio spectrum in this paper. The TV band is divided into M channels, and the activity of PU is modeled as an exponential on/off activity pattern. The following binary hypothesis model defines how each vehicle and ship senses the spectrum:

$$x_i(t) = \begin{cases} n_i(t), & H_0 \\ s_i(t) + n_i(t), & H_1 \end{cases} \tag{4}$$

where $n_i(t)$ is additive white Gaussian noise, $s_i(t)$ is the primary signal received by the vehicle/ship i, and $i = 1, 2, ..., VS$, with VS as the total number of vehicles and ships in the integrated coastal city. Each vehicle and ship computes its local observation about the spectrum as an a posteriori probability, which is calculated as:

$$\varphi_i^f(H_h) = P(H_h|x_i) = \frac{P(x_i|H_h)P(H_h)}{P(x_i)} \tag{5}$$

where f represents the channel index—that is, like how [21], $f \in M$, $P(x_i|H_h)$ is the probability density function of the normally distributed random variable x_i conditioned on H_h, $(h = 0, 1)$.

$P(x_i)$ is a normalizing constant, and $P(H_h)$ is the a priori probability that is supposed to be constant for all vehicles and ships. To compute the belief about the condition of the vehicle/ship j estimated by vehicle/ship i, vehicles/ships i and j within the transmission range of one another combine their messages in the following manner:

$$\mu_{ij}^f(H_j) = w \sum_{H_i} \psi_{ij}^f(H_i, H_j) \times \varphi_i^f(H_i) \prod_{k \in (N_i - \{j\})} \mu_{ki}^f(H_i) \tag{6}$$

$\mu_{ij}^f(H_j)$ defines the belief about the vehicle/ship j's current condition estimated by vehicle/ship i; w is the weight factor, and the term $k \in (N_i - j)$ explains that k only considers the neighbors of i and it does not belong to the neighbors of j, and the compatibility function, $\psi_{ij}^f(H_i, H_j)$, is calculated as:

$$\psi_{ij}^f(H_i, H_j) = \begin{cases} \eta & \text{if } H_i = H_j \\ 1 - \eta & \text{if } H_i \neq H_j \end{cases} \tag{7}$$

Each vehicle and ship eventually computes the belief as:

$$b_i^f(H_i) = w\, \varphi_i^f(H_i) \prod_{k \in (N_i)} \mu_{ki}^f(H_i) \tag{8}$$

Based on these beliefs, each moving node then makes a final decision about the current state of the channel, as follows:

$$D_i^f = \begin{cases} H_0 & \text{if } b_i^f(H_0) > b_i^f(H_1) \\ H_1 & \text{if } b_i^f(H_0) < b_i^f(H_1) \end{cases} \tag{9}$$

Now, the term $min(Ch_1, Ch_2, ..., Ch_M)$ in (3) describes how a querying node selects the channel with the highest belief from the set of all the idle channels between two communicating nodes. The term Ch_{f^*} for the idle channel $f^*(f^* = 1, ..., M)$ between vehicle/ship i and vehicle/ship j is calculated as $Ch_{f^*} = 1 - min\left(b_i^{f^*}(H_0), b_j^{f^*}(H_0)\right)$. Hence, the path is selected by choosing, hop-by-hop, the highest Q-values for each link between the source and destination. The lower the value of TD, the higher the reward. In order to make the convergence faster in a mixed (terrestrial plus marine) environment and to reduce latency, we used (1) to update the Q-values.

To generate a marine environment for an integrated coastal city, we considered the similar division as described by Pierson Jr. and Moskowitz [22]. They divided sea states into 10 different levels and measured each level by using a significant wave height, wave period, and wave length as parameters. Due to the continuous movement of the sea surface, different ships receive different power levels of the received signal, thereby resulting in different signal-to-noise ratios (SNRs), which is defined as:

$$\gamma_i = \frac{P_{R_i}}{N_o W} \tag{10}$$

where $N_o W$ represents the total noise power, and P_{R_i} is the power received at each ship i, which is calculated as follows:

$$P_{R_i} = P_T - PL \tag{11}$$

P_T and PL are the transmitted power and path loss, respectively. For the sea environment, path loss PL is a function of frequency f and sea surface height h_s, and was defined by Timmins and O′Young [23] as follows:

$$PL(h_s, f) = PL(d_o) + 10 + [(0.498\, log_{10}(f) + 0.793)h_s + 2] \\ log_{10}\left(\frac{d}{d_o}\right) + X_f \tag{12}$$

$PL(d_o)$ defines the path loss model which is measured from the transmitter at a reference distance d_o, and X_f is a Gaussian random variable with zero mean and standard deviation, given as $\sigma_f = [0.157f + 0.405] \times h_s$.

Figure 3 presents the complete algorithm by describing in what way the user benefits from the integrated cognitive coastal city network. The nodes (moving vehicles and ships) periodically exchange messages with each other and with the corresponding LCs in order to maintain an updated network state. The exchanged information is their IDs, positions, velocities, and channel states. To keep from discrimination among different communicating nodes, we used the v_ID, v_position, v_velocity, and v_channel for vehicular communications, and s_ID, s_position, s_velocity, and s_channel for marine communications. The channel state represents the existence and nonexistence of the PU in the network. This updated information is exchanged between LCs and the SDN controller, respectively, so that an updated local and global network topology is maintained. Accordingly, whenever a user demands a service, it sends an SRM to an LC. The LC replies with an SPM if it has a route to the destination of the querying node; otherwise, it asks the SDN controller for the SPM. The SPM in this paper is a stable route calculated with (1). The SDN controller sends the SPM back to the LC if it has a route for the querying node; otherwise, it requests the service from the corresponding VNFs if the VNF for that service was already launched by NFV. If not, it sends the SRM to the NFV to launch the corresponding VNF and provide the service.

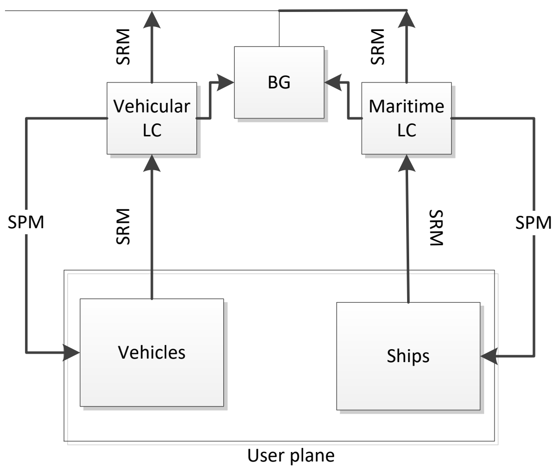


Figure 3. A flowchart representing the proposed architecture.

3. Simulation Results and Discussion

We simulated our proposed scheme in NS-2 to evaluate the network performance for one vehicular source-destination pair and one maritime source-destination pair. The total number moving in the network (VS) varied between 6 and 24, each with a communication range of 200 m. Vehicles and ships moved at varying speeds up to a maximum of 15 m/s. The LC for the marine network moved at a speed of 10 m/s close to shore to keep it connected with the SDN controller. We divided the spectrum into $M = 5$ bands where any channel could be occupied by a licensed PU. Two PUs were used in the network, each with a communication range of 500 m. These PUs were fixed nodes which considered an exponential on/off activity pattern with a rate parameter of 0.05. There were two RSUs, each with a communications range of 350 m. The fog cloud was a single fixed node that acted as a central controller for the simulations, with $\beta = 0.7$, $\gamma = 0.8$, $P_{ss} = 0$, $P_{ss'} = 1$, and $\eta = 0.9$. For generating the marine environment, the path loss model defined in (12) was considered for the simulation. A moderate sea state with a wave height between 1.83 m and 2.29 m [22] was used for the stability between the two environments. These simulation parameters are listed in Table 1.

Table 1. Simulation parameters.

Parameters	Values
number of vehicles and ships, VS	between 6 and 24
communication range of each VS	200 m
speed of VS	up to 15 m/s
speed of marine LC	10 m/s
number of channels, M	5
number of PUs	2
communication range of PU	500 m
rate parameter of exponential on/off activity	0.05
number of RSUs	2
communication range of RSUs	350 m
β	0.7
γ	0.8
η	0.9
P_{ss}	0
P'_{ss}	1
wave height	between 1.83 ans 2.29

As mentioned above in Section 1, the literature has no publicly recognized integrated coastal city combining cognitive vehicular and maritime communications to provide a routing service for different users' demands. Therefore, we first evaluated our proposed scheme for V2V, S2S, and V2S communications in the integrated framework, one by one, and then chose to compare the V2V and S2S communications of our proposed scheme with the hierarchical software-defined VANET (HSDV) [24] routing protocol. To evaluate the impact of our proposed integrated scheme in the respective environments, we only compared V2V and S2S with their respective reference schemes. In HSDV, the controller selects the relay nodes to make stable links from the source to the destination by computing the distance with reference to the destination. On the other hand, our integrated architecture chooses the path by selecting the maximum reward value, hop-by-hop, where the reward value is based on less transmission delay and high connectivity among the neighboring nodes. The second reason to choose HSDV as a reference scheme is the deployment of local controllers. Both our proposed scheme and HSDV consider local controllers to divide the burden of the main controller and to provide the local view of the network to the main controller. Moreover, the vehicular and maritime networks are concerned with topological constraints, and a ship at sea is analogous to a vehicle in traffic; therefore, we evaluated HSDV in the marine environment just for the sake of comparison. For simplicity, we used the fog cloud as a central controller for the proposed scheme, which provides service according to the users' demands by communicating with the corresponding

VNF. Likewise, in our previous work [16], we simulated HSDV with a spectrum-sensing scheme for vehicular networks proposed by Abbassi et al. [25] to make it cognitive HSDV, and referred to it as the reference scheme for V2V. Similarly, for cognitive maritime networks, we simulated HSDV with the spectrum selection scheme proposed by Tang et al. [26] and named it as the reference scheme for S2S. We analyzed our integrated coastal city with three different metrics:

(*a*) packet delivery ratio
(*b*) end-to-end delay
(*c*) routing overhead ratio

Figure 4 presents the packet delivery ratio with the number of nodes as an independent variable for different communications systems, and with different idle probabilities of the PU as a parameter. The packet delivery ratio is a measure of the number between the delivered generated packets. The delivery ratio increases for all the communications systems when the number of nodes (vehicles and ships) in the network increases. The pattern is similar for hybrid and pure communications—that is, the network shows better performance with an increasing number of nodes and increasing idle probability of the PU together. This is because increasing the number of nodes in the network increases the connectivity, and in the second case, the high probability means that the channel is free from the PU for a long time, which provides more common free channels in the network. Moreover, the fog cloud improves network performance in terms of packet delivery ratio, since all the functions and the network management are now centrally controlled by the controller that increases the network intelligence by globally dictating network behavior. The word *globally* here means that the controller is responsible for keeping the global network view by interacting with LCs. This means that the controller manages the entire information about the status of channels and relays. As a result, by computing the maximum reward, the controller is responsible for providing the more stable paths from source to destination per inquiring node. When the network is sparse at a 70% idle probability for the PU, we achieved 69% successful delivery of packets for S2S, 79% for V2V, and 80% for V2S communications. The delivery ratio of V2S communications is high, compared to the other two communications systems when the number of nodes is high. This is because of the fog cloud that maintains stability between the two environments. The S2S performance is poor because of the constantly changing sea surface. Figure 4 also shows that when the idle probability is low, there is a performance degradation in the network. This is because with low probability, nodes face difficulty in reaching a consensus on a common free channel.

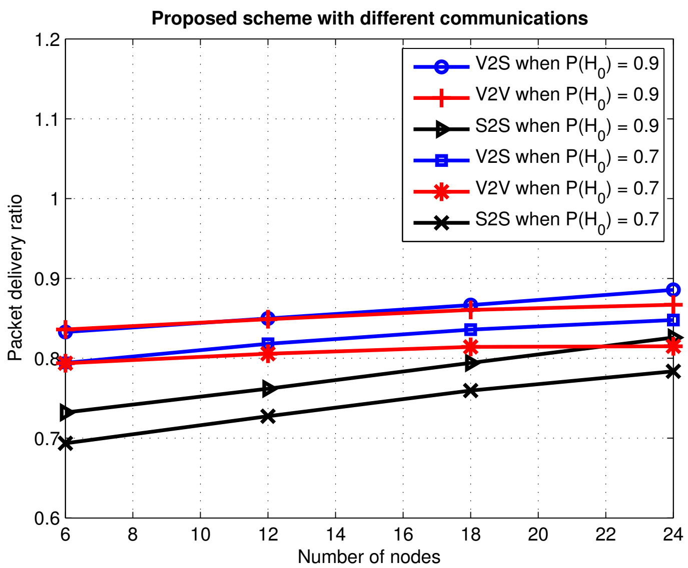


Figure 4. Performance comparison between the proposed hybrid and pure communications in terms of packet delivery ratio.

Figure 5 shows the end-to-end delay with the number of nodes as an independent variable for different communications systems, and with different idle probabilities of the PU as a parameter. End-to-end delay is a measure of the difference between the number of times a packet starts going from a source and ends at the destination. When the network is sparse between (6 and 12 nodes), the delay is high for all the communications. The independent variable, that is, the number of nodes in the network decreases the delay with an increase in its numeral. This pattern of improved performance in terms of delay with an increasing number of nodes and increasing idle probability of the PU, is again the same for both hybrid and pure communications. This is because increasing the number of nodes in the network increases the connectivity, and in the second case, the high probability means that the channel is free from the PU for a long time, which provides more common free channels in the network. The delay is high for S2S communications because of the environmental factor. This means that with the changing positions of ships in a marine environment, the sea surface is also changing constantly and, therefore, results in more fragile links than in terrestrial networks. Moreover, hybrid communications shows better performance due to its hybrid nature, that is, the path is divided into two types of networks. For that reason, the SDN controller is responsible for providing more stable paths with the help of the BG. Figure 5 also demonstrates that when increasing the idle probability of the PU, the performance improves for all the communication systems. The reason for this is similar to that described above—the higher probability increases the chances of free channels. However, with a decrease in idle probability, the chances among the nodes to show stability also decreases, because low probability causes difficulty for them to have a consensus about a common free channel. This is because a querying node may find another node when the idle probability is low, but it may not choose that one as a relay node due to not reaching a consensus on a common channel, thereby decreasing the chances for nodes to make stable links among them. Consequently, the nodes in the network do not provide better performance, as they do not reach a consensus on a common free channel when the idle probability of the PU is low.

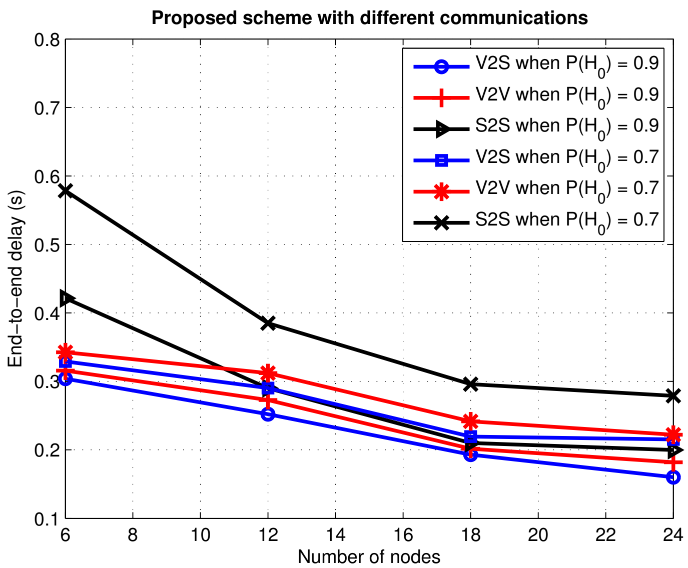


Figure 5. Performance comparison between the proposed hybrid and pure communications in terms of end-to-end delay.

Figure 6 shows the routing overhead ratio when the number of vehicles and ships in the network is increased for different communications systems, and when the idle probabilities of the PU is also changed. The routing overhead ratio is a measure of the amount of control packets from the entire amount of packets in the network. From the Figure, we can see that by increasing the number of nodes in the network and decreasing the idle probability of the PU, the overhead ratio increases for each of these scenarios. Nevertheless, the V2S communications performs better than the other two communications systems in terms of network overhead ratio. This is because the BG and LCs reduce the chances of link fragility, thereby decreasing the amount of control messages in the network. Furthermore, choosing the best stable path from the source to destination at the controller with the maximum reward lowers the overall network overhead. Each querying node sends a request to the controller each time it comes across a mismatch in the packet fields or entails an updated route. Accordingly, with an increase in the number of nodes, the update rate of messages in the network is also increased. The high update rate for fixed nodes is due to not finding a free channel. Figure 6 also shows that a decrease in the idle probability of the PU degrades the network performance in terms of the overhead ratio. This is because the long active state of the PU does not allow for facilitation of any secondary user. Therefore, the chances of finding a common free channel among nodes become fewer, and accordingly, the high exchange rate of messages among nodes degrades the network performance.

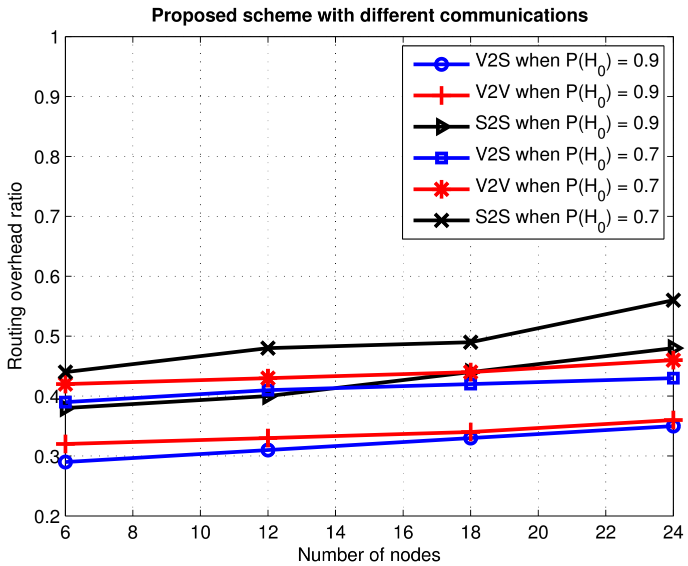
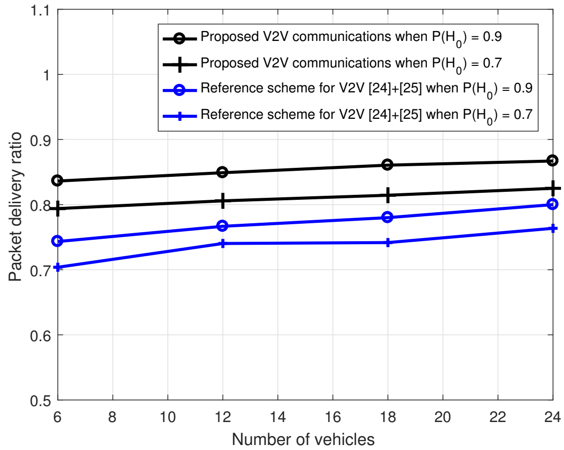
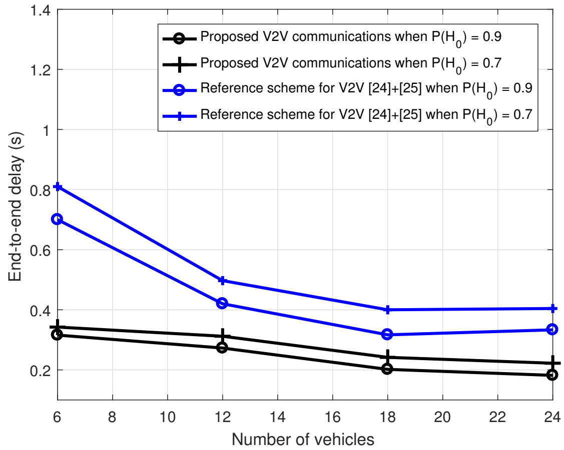


Figure 6. Performance comparison between the proposed hybrid and pure communications in terms of routing overhead ratio.

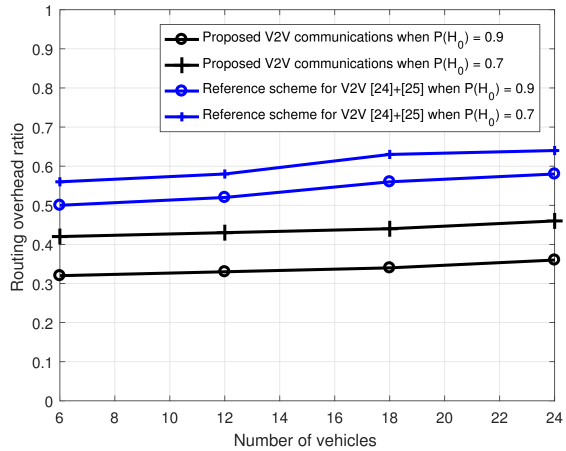
An additional set of results is demonstrated in Figures 7 and 8. Figure 7a–c, respectively show the packet delivery ratio, end-to-end delay, and overhead ratio of the two cognitive vehicular schemes. Each of these figures were evaluated by varying the idle probabilities of the PU with a different number of vehicles. Similarly, Figure 8a–c, respectively show the packet delivery ratio, end-to-end delay, and overhead ratio of the two cognitive maritime schemes, which were evaluated by varying the number of ships in the network and by changing the idle probabilities of the PU. Our proposed schemes show improved performance than the reference schemes in both vehicular and maritime environments. The first reason is the difference in choosing the next relay to make stable paths from the source to destination. The reference schemes select the nearest vehicle or ship to the destination by measuring the distance between nodes, which merely creates intermittent link connections, thereby decreasing the chances of successfully delivered packets. For the same reason, the reference schemes show poor performance under sparse network conditions. However, our proposed scheme calculates rewards based on transmission delays that include both the speed and direction of vehicles and ships. Secondly, for fewer nodes, the reference schemes do not ensure stability in cognitive vehicular and maritime networks. With a decrease in the probability of PU being idle, the chances of having a common free channel between nodes also decreases. This means that our proposed schemes show stability even when the number of vehicles and ships is less, and when the network is a cognitive network. A comprehensive review of our simulation results is that network performance varies with the type of communication system, the type of environment (terrestrial or marine), as well as the type of relay node.



(**a**)

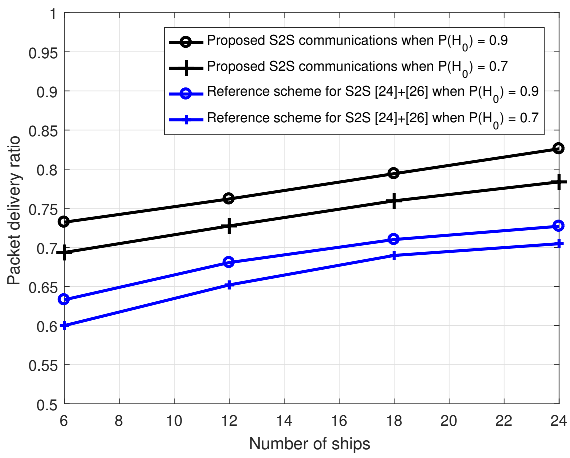


(**b**)

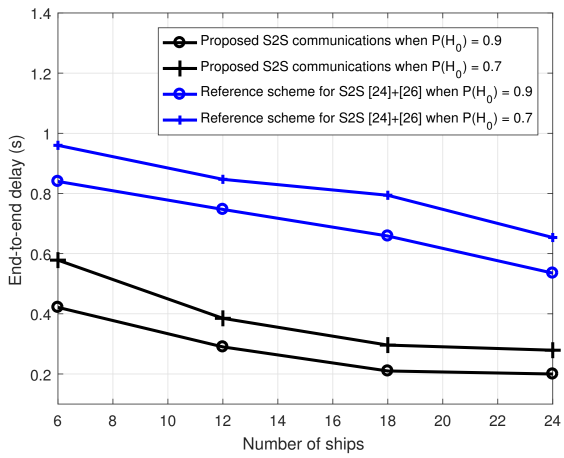


(**c**)

Figure 7. Performance comparison between the proposed V2V and the reference scheme for V2V in terms of (**a**) packet delivery ratio, (**b**) end-to-end delay, and (**c**) routing overhead ratio.

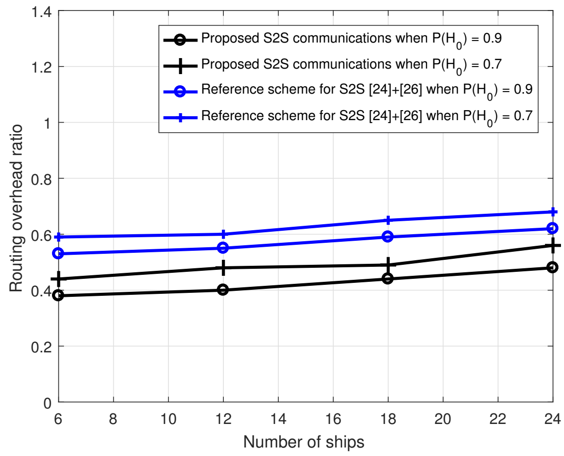


(**a**)



(**b**)

Figure 8. *Cont.*



(**c**)

Figure 8. Performance comparison between the proposed S2S and the reference scheme for S2S in terms of (**a**) packet delivery ratio, (**b**) end-to-end delay, and (**c**) routing overhead ratio.

4. Conclusions

This paper proposed a new integrated cognitive coastal city for both hybrid and pure communications between vehicular and maritime networks. The idea of integrating software-defined networking, network function virtualization, and fog computing in a cognitive routing scheme for different networks makes this integrated framework unique. Vehicle-to-ship communications was introduced to provide efficient and reliable services to users on different network systems. Two virtual networks for each vehicular and maritime application were launched by NFV based on the user demands, and the SDN controller was responsible for controlling and managing these VNs. Routing is the service provided by the VNs, where the SDN controller provides a stable path between each source-destination pair for each querying node. These pairs are vehicle-to-ship, vehicle-to-vehicle, and ship-to-ship. The border gateway is used as a communicating device between the two networks. The paths were established for communication between different cognitive networks using this integration. The idea was to facilitate citizens' demands for services from different applications under the same infrastructure.

Our results show better performance in hybrid communications for packet delivery ratio, end-to-end delay, and routing overhead ratio. We considered both pure and hybrid communications in this scheme, and concluded that the marine environment has more fragile links than the terrestrial environment. However, for hybrid communications, we achieved better performance in terms of delivery ratio, delay, and overhead, since the path was divided into two types of networks and border gateways, and local controllers reduce the chances of link fragility. Hence, the demand of each application user in a hybrid environment can be fulfilled with a quicker response time via the proposed scheme. In comparison with the existing schemes in literature, our scheme performs better, due to the following reasons: Firstly, it is because of our selection criterion that depends on the transmission delay, and secondly, because of the stability that our scheme shows due to the integration of software-defined networking, network function virtualization, and fog computing. We will extend the applications for both vehicular and maritime networks along with different services in the near future.

Author Contributions: Conceptualization, H.G. and I.K.; formal analysis, H.G.; methodology, H.G. and I.K.; Writing—Original draft preparation, H.G.; Writing—Review and editing, I.K.; supervision, I.K.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2018R1A2B6001714).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Baktir, A.C.; Ozgovde, A.; Ersoy, C. How can Edge Computing Benefit from Software-Defined Networking: A Survey, Use Cases, and Future Directions. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 2359–2391. [CrossRef]
2. Bu, C.; Wang, X.; Cheng, H.; Huang, M.; Li, K.; Das, S.K. Enabling Adaptive Routing Service Customization via the integration of SDN and NFV. *J. Netw. Comput. Appl.* **2017**, *93*, 123–136. [CrossRef]
3. Jiang, D.; Delgrossi, L. IEEE 802.11p: Towards an international standard for wireless access in vehicular environments. In Proceedings of the VTC Spring 2008-IEEE Vehicular Technology Conference, Singapore, 11–14 May 2008; pp. 2036–2040.
4. Zhou, M.-T.; Hoang, V.; Harada, H.; Pathmasuntharam, J.S.; Wang, H.; Kong, P.Y.; Ang, C.W.; Ge, Y.; Wen, S. TRITON: High-speed maritime wireless mesh network. *IEEE Wirel. Commun.* **2013**, *20*, 134–142. [CrossRef]
5. Bonfim, M.S.; Dias, K.L.; Fernandes, S.F.L. Integrated NFV/SDN Architectures: A Systematic Literature Review. *arXiv* **2018**, arXiv:1801.01516.
6. Alcatel.Lucent. *The Right SDN is Right for NFV*; Strategic White Paper; September 2014.
7. Pan, J.; Ma, L.; Ravindran, R.; TalebiFard, P. HomeCloud: An Edge Cloud Framework and Testbed for New Application Delivery. In Proceedings of the International Conf. on Telecommunications (ICT), Thessaloniki, Greece, 16–18 May 2016.
8. Ying, H.; Yu, R.; Zhao, N.; Leung, V.C.M.; Yin, H. Software-Defined Networks with Mobile Edge Computing and Caching for Smart Cities: A Big Data Deep Reinforcement Learning Approach. *IEEE Commun. Mag.* **2017**, *55*, 31–37.
9. Zhang, W.; Zhang, Z.; Chao, H.-C. Cooperative Fog Computing for Dealing with Big Data in the Internet of Vehicles: Architecture and Hierarchical Resource Management. *IEEE Commun. Mag.* **2017**, *55*, 60–67. [CrossRef]
10. Yang, P.; Zhang, N.; Bi, Y.; Yu, L.; Shen, X. Catalyzing Cloud-Fog Interoperation in 5G Wireless Networks: An SDN Approach. *IEEE Netw.* **2017**, *31*, 14–20. [CrossRef]
11. Feng, B.; Zhou, H.; Zhang, H.; Li, G.; Li, H.; Yu, S.; Chao, H.C. HetNet: A Flexible Architecture for Heterogeneous Satellite-Terrestrial Networks. *IEEE Netw.* **2017**, *31*, 86–92. [CrossRef]
12. Wang, K.; Yin, H.; Quan, W.; Min, G. Enabling Collaborative Edge Computing for for Software Defined Vehicular Networks. *IEEE Netw.* **2018**, *32*, 112–117. [CrossRef]
13. Liang, K.; Zhao, L.; Chu, X.; Chen, H.-H. An Integrated Architecture for Software Defined and Virtualized Radio Access Networks with Fog Computing. *IEEE Netw.* **2017**, *31*, 80–87. [CrossRef]
14. Campolo, C.; Molinaro, A.; Iera, A.; Menichella, F. 5G Network Slicing for Vehicle-to-Everything Services. *IEEE Wirel. Commun.* **2017**, *24*, 38–45. [CrossRef]
15. Hu, T.; Fei, Y. QELAR: A Machine-Learning-Based Adaptive Routing Protocol for Energy-Efficient and Lifetime Extended Underwater Sensor Networks. *IEEE Trans. Mobile Comput.* **2010**, *9*, 796–809.
16. Ghafoor, H.; Koo, I. CR-SDVN: A Cognitive Routing Protocol for Software-Defined Vehicular Networks. *IEEE Sens. J.* **2018**, *18*, 1761–1772. [CrossRef]
17. Ghafoor, H.; Koo, I. Infrastructure-aided hybrid routing in CR-VANETs using a Bayesian Model. *Wirel. Netw.* **2019**, *25*, 1711–1729. [CrossRef]
18. Yedidia, J.S.; Freeman, W.T.; Weiss, Y. Understanding belief propagation and its generalizations. In *Exploring Artificial Intelligence in the New Millennium*; Lake-meyer, G., Nebel, B., Eds.; Morgan Kaufmann: San Francisco, CA, USA, 2001; pp. 2282–2312.
19. Baraka, K.; Safatly, L.; Artail, H.; Ghandour, A.; Hajj, A.E. An infrastructure-aided cooperative spectrum sensing scheme for vehicular ad hoc networks. *Ad Hoc Netw.* **2015**, *25*, 197–212. [CrossRef]
20. Yang, T.; Feng, H.; Yang, C.; Sun, Z.; Yang, J.; Sun, F.; Deng, R.; Su, Z. Cooperative Networking towards Maritime Cyber-Physical Systems. *Int. J. Distrib. Sens. Netw.* **2016**, *12*, 3906549. [CrossRef]
21. Ghafoor, H.; Noh, Y.; Koo, I. Belief propagation-based cognitive routing in maritime ad hoc networks. *Int. J. Distrib. Sens. Netw.* **2016**, *12*, 7635206. [CrossRef]
22. Pierson, W.J., Jr.; Moskowitz, L. A proposed spectral form for fully developed wind seas based on the similarity theory of SA Kitaigorodskii. *J. Geophys. Res.* **1964**, *69*, 5181–5190. [CrossRef]
23. ITimmins, J.; O'Young, S. Marine communications channel modeling using the finite-difference time domain method. *IEEE Trans. Veh. Technol.* **2009**, *58*, 2626–2637. [CrossRef]

24. Correia, S.; Boukerche, A.; Meneguette, R.I. An Architecture for Hierarchical Software-Defined Vehicular Networks. *IEEE Commun. Mag.* **2017**, *55*, 80–86. [CrossRef]
25. Abbassi, S.H.; Qureshi, I.M.; Abbasi, H.; Alyaie, B.R. History-based spectrum sensing in CR-VANETs. *EURASIP J. Wirel. Commun. Netw.* **2015**, *2015*, 163. [CrossRef]
26. Tang, C.; Kandeepan, S.; Hourani, A.; Munari, A.; Berioli, M. Spectrum sensing for cognitive maritime VHF networks. In Proceedings of the 2014 MTS/IEEE Taipei Conference: Oceans Regeneration, OCEANS′14, Taipei, Taiwan, 7–10 April 2014.