

Effective Heuristics for Flexible Spectrum Access in Underwater Acoustic Networks

Nicola Baldo*, Paolo Casari*, Paolo Casciaro* and Michele Zorzi*[†]

*Department of Information Engineering — University of Padova — Via Gradenigo 6/B, I-35131 Padova, Italy

[†]California Institute of Telecommunications and Information Technology – UC San Diego, USA

E-mail: {baldo, casarip, pcasciar, zorzi}@dei.unipd.it

Abstract—In this paper, we consider underwater networks with multiple sensors reporting data to a single sink node, and advocate the use of FDMA as a channel access method. In order to assign channels to users in an efficient way, we compare a number of allocation methods that explicitly account for the underlying channel attenuation and noise model. We begin by inspecting a simpler case, where the number of channels, N_{ch} , is equal to the number of users, N_u , and shortly describe a known algorithm which finds an optimal channel allocation in a max-min fair sense. We then assume that $N_{ch} > N_u$, and explore how the channel assignment problem changes. We finally propose suitable heuristic methods that allow to find effective solutions without the complexity of an exhaustive exploration of all possible assignments.

Performance evaluation by means of simulation shows that our heuristic methods maintain fairness among users, while achieving a better minimum performance than optimal methods for the $N_{ch} = N_u$ case. We also show when it is more effective to use a given allocation method as a function of the network area extension. Finally, we compare our FDMA-based allocation method with the ALOHA random access control mechanism, showing how our channel allocation can improve the overall throughput of the network.

Index Terms—Acoustic communications, underwater networking, dynamic spectrum access, frequency-division multiple access.

I. INTRODUCTION AND RELATED WORK

UNDERWATER communications pose different problems to networking protocol design than terrestrial radio networks. The main discrepancies between the two scenarios lie in the very low propagation speed, five orders of magnitude smaller than the speed of radio waves in the air, and in the frequency-dependent channel attenuation and noise power spectral density, which cause the average power received over any link to show a dependence on frequency as well. It should be noted that these two specific problems are quite well known in the networking community. Specifically, long propagation delays are typical of very-long range radio communications, especially involving satellite links [1] or, e.g., deep space probe transmissions. Furthermore, frequency-dependent channel effects have been studied ever since the early years of copper wireline communications [2]. However, these problems were usually considered separately, and dealt with using techniques that specifically depend on the scenario and application to serve.

In underwater communications, these problems are to be addressed jointly, and with completely different objectives than faced in past research on radio transmissions. Underwater network nodes face challenges at a higher and broader level, as they will be required to abide to the ad hoc paradigm, and therefore to organize themselves and perform any other operation, including communication, in a completely independent way. For this reason, long delays and frequency-dependent effects need be taken into account at an early design stage and be incorporated within access techniques and protocols for underwater networks.

For example, Time-Division and Carrier-Sense Multiple Access schemes (TDMA and CSMA), while successful in traditional radio communications, are prone to severe limitations in efficiency and scalability when employed in the underwater environment. In either case, such disadvantages are primarily due to the very large propagation delays. On one hand, TDMA schedules require that each slot be occupied by the transmission of only one node: thus, each slot must contain a guard time long enough to cover the maximum propagation time of the network. While TDMA can be of some use in clustered topologies for intra-cluster communications (because clusters usually have a limited geographical extension) [3], [4], in case of larger networks TDMA would lead to very long guard times and hence high inefficiency, which comes in addition to synchronization overheads and the already low transmit rates typical of underwater channels.

On the other hand, CSMA works well only if the propagation time on the channel is much smaller than the duration of a data packet transmission: in typical underwater scenarios, with the long propagation delays cited before, this is usually not the case. Augmenting CSMA with collision avoidance schemes, such as Request-To-Send / Clear-To-Send (RTS/CTS) handshakes, is prone to the same inefficiency that affects CSMA [5]: again, long delays make collisions between packets more likely and frequent. This may lead to prefer simpler protocol, such as ALOHA [6], which is however inherently inefficient.

As a consequence of TDMA's inherent inefficiency, all TDMA-based (in general, time-scheduled) access methods usually bear low data throughput. For example, Slotted FAMA [7] relies on time-slotted RTS/CTS exchanges prior to data transmissions, whereas UWAN-MAC's [8] schedules transmit and receive epochs according to an awake/sleep schedule in order to let nodes save some energy while ensuring connectivity. Both protocols can be shown to achieve low throughput, for different reasons: Slotted FAMA is easily out-

performed by non-time-synchronous protocols that try to limit collisions instead of avoiding them completely [9]; UWAN-MAC can operate only on very low duty cycles (i.e., ratios of awake time over total time) and has additional overheads due to signaling messages used to synchronize transmit schedules [10].

To summarize, the major issues with MAC schemes which operate entirely in the time domain are limited efficiency and poor scalability due to the huge underwater propagation delay. For this reason, FDMA becomes very attractive, as its efficiency is not affected by long propagation delays.¹

However, the use of FDMA in underwater communications requires to handle the propagation effects peculiar of the underwater channel. Due to frequency-dependent attenuation and noise, the capacity achievable on a particular frequency band strongly depends both on frequency itself and on the communication distance, unlike in traditional radio transmissions where different FDMA channels usually have comparable performance. Therefore, fixed channel allocation schemes traditionally used for radio FDMA do not perform well in underwater environments. In turn, this calls for new algorithms that efficiently allocate channels to transmitters while explicitly accounting for the specific features of underwater propagation.

In this paper, we advocate the use of FDMA as a deterministic resource allocation scheme, and compare a number of channel allocation methods that make it suitable for use in the underwater environment. We begin by inspecting a simpler case, where the number of channels N_{ch} is equal to the number of users N_u , and briefly recall the description of an algorithm which finds an optimal channel allocation in a max-min fair sense. We then argue that the minimum channel capacity among all users could be improved by increasing the degrees of freedom of the algorithm, that is, by varying the ratio of the number of channels to the number of users $N_{ch}/N_u = k$ for some $k > 1$. Since the allocation problem becomes NP-hard in this case [12], we deploy suitable heuristic methods that are less complex, yet can effectively improve the minimum capacity achieved by all nodes in the network.

Performance evaluation by means of simulation shows that fairness among users is maintained with respect to the simpler $N_{ch} = N_u$ case, while achieving better minimum performance. We also show when it is more effective to use a given allocation method as a function of the network area extension. Finally, we compare our allocation method with the ALOHA random access control mechanism, showing how our channel allocation can improve the overall throughput of our network.

II. OVERVIEW OF CHANNEL EFFECTS

Any decision made by algorithms for channel allocation in underwater FDMA must properly take into account channel effects and their consequences. This section briefly summarizes these effects before proceeding to introducing and commenting allocation algorithms.

It has been shown that the underwater channel exhibits frequency-dependent attenuation and noise [13]. More specifically, the underwater attenuation $A(d, f)$ can be expressed as

$$A(d, f) = d^k a(f)^d, \quad (1)$$

where d is the distance between transmitter and receiver, and k is the counterpart of the path loss coefficient in terrestrial radio, and is used to model the geometry of propagation. A practical value $k = 1.5$ is usually adopted. The factor $a(f)$ in (1) is called the absorption loss: it models the conversion of acoustic pressure into heat, and can be approximated by Thorp's formula [14], [15].

The noise power spectral density (psd), $N(f)$, is also frequency-dependent, and is usually expressed as a superposition of four contributions: turbulence, shipping and other human activities, wind and waves, and thermal noise in the receiver circuitry. These components contribute to the noise psd at different frequencies of the acoustic spectrum, and can be modeled as reported in [15]. Assume to transmit a pure acoustic tone. Once the transmit power, distance and frequency are known, the signal-to-noise ratio (SNR) of the received tone can be modeled as

$$SNR(d, f) = \frac{P_T}{A(d, f)N(f)\Delta f}, \quad (2)$$

where P_T is the transmit power, and Δf is a narrow band around f . In (2), the factor $[A(d, f)N(f)]^{-1}$ (also called the AN factor in the sequel) is the frequency-dependent term. It should be noted that $A(d, f)$ increases with frequency while $N(f)$ decreases (at least to a certain point). Hence, the inverse of the product of the two factors has a maximum for some frequency f_0 . This maximum represents the best frequency to use to transmit the tone [15].

A closer observation of the AN factor gives the best insight about the behavior of allocation policies. To this end, in Figure 1 we plot the term $[A(d, f)N(f)]^{-1}$ in (2), for a number of values of d . Each gray line corresponds to a different distance; some relevant distance values are labeled for illustration. This figure is inspired to similar figures in [15], [16], and first of all shows that there is in fact an optimal frequency f_0 that offers the best conditions to transmit a tone at a given distance. Figure 1 also reports a simpler version of a channel allocation problem, where a given spectrum is split between two users, which are assigned one of the two channels that constitute the overall system bandwidth. We assume that both users have to transmit to a common receiver (the sink), and that user 1 is 1 km away, whereas user 2 is located at a greater distance, namely 5 km from the sink. The system bandwidth spans from 10 kHz to 40 kHz, and is divided into two equally wide channels of 15 kHz each. We observe that two different resource allocations are possible: *i*) user 1 on channel 1 and user 2 on channel 2, or *ii*) vice-versa. According to the chosen allocation, each user experiences a different channel response, due to distance- and frequency-dependent propagation effects. This is highlighted by a bold solid line and a bold dash-dot line that correspond to the channel responses undergone by the users in cases *i*) and *ii*), respectively.

¹It should be noted that FDMA has already been successfully demonstrated in a real deployment (e.g., see [11]).

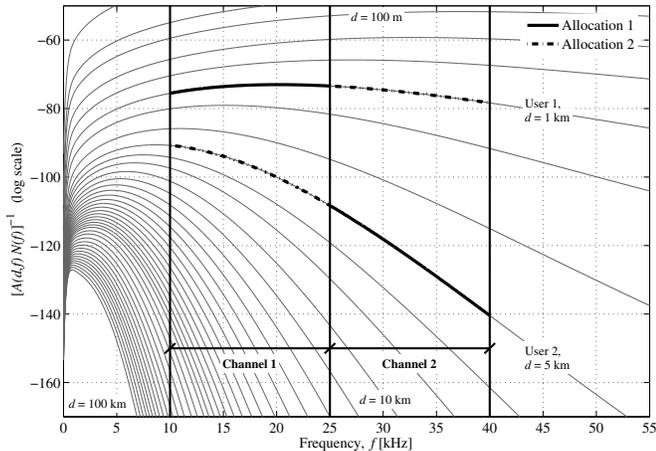


Figure 1. Frequency-dependent part of the SNR for an acoustic tone transmitted underwater, and example of band allocation to two users, one at 1 km, the other at 5 km. Grey lines represent the $[A(d, f)N(f)]^{-1}$ factor for different distances. The bold black solid and dot-dashed lines represent the non-optimal and optimal allocation, respectively [16].

Assume that both users transmit with the same power: therefore user 2 experiences the minimum throughput, because of the larger distance and worse channel effects. If we want to maximize the minimum throughput experienced by all users, our attention must then be concentrated on user 2. To this end, allocation 1 (solid line) is not optimal, because combined attenuation and noise effects severely limit user 2's experienced performance. On the contrary, allocation 2 (dash-dot line) is optimal. Furthermore, a general concept that can be applied here, and must be taken into consideration by any allocation algorithm, is that longer links (i.e., users located farther from the sink) are typically more constrained than shorter links, and should be privileged when allocating resources; moreover, allocating lower-frequency channels to these users usually corresponds to allowing more favorable propagation effects, thus to improving minimum network performance. These facts uniquely depend on the propagation characteristics of the underwater acoustic channel, whereas in terrestrial radio all users would experience the same long-term channel effects in all sub-bands, hence making any frequency allocation equally good.

In the following, we will assume that the throughput of a user on a certain channel is equal to the channel capacity. Given the above characterization, and assuming that f_ℓ and f_r are the lower and upper frequencies of a specific channel used for communication, the channel capacity can be calculated according to the Shannon-Hartley theorem as

$$\int_{f_\ell}^{f_r} \log_2 \left(1 + \frac{S(f)}{N(f)} \right) df, \quad (3)$$

where $S(f)$ is the spectrum of the signal to be transmitted. Note that (3) can be applied to any signal or noise spectrum.

III. DESCRIPTION OF ALLOCATION ALGORITHMS

In the following, we will consider the FDMA channel allocation problem as applied to single-hop networks, with a sink node acting as a common receiver. We assume that the network

is formed of N_u nodes that are located at different distances from the sink, and as such undergo different propagation effects. For now, assume also that a prescribed (and fixed) portion of the acoustic spectrum, i.e., the system bandwidth, is split in N_{ch} channels, each to be assigned to exactly one user. We recall that the optimality of a user-channel association depends on capacity, which in turn depends on distance: thus, the most important net effect is that a high-frequency channel is more suited to a user located close to the sink, whereas a low-frequency channel yields better performance to farther users. All algorithms we describe are based on these concepts. However, due to the particular shape of the attenuation-noise function, it is not possible in general for each user to be assigned its optimal channel. Furthermore, transmit power limitations usually give rise to near-far effects, whereby closer nodes experience better channel capacity. This effect is even more important in underwater networks, where distance-related effects enhance the available transmission bandwidth for near users. Therefore, we need to abide to a general optimality criterion that allows to enhance the capacity of far users (that will have low to very low capacities), without sacrificing too much the capacity of nearer users. We also note that, in practical scenarios, a near user will achieve a fairly high capacity even when using a sub-optimal channel, whereas a far user may experience very low capacity unless it is assigned a sufficiently low-frequency channel.

The algorithms we describe in the sequel leverage on these opportunities to achieve *max-min fairness* [17] among the users, which entails the concept of iteratively allocating resources so that the minimum performance experienced among all users is maximized at every optimization step.

A. Optimal max-min fair allocation in the case $N_u = N_{ch}$ (CFDMA)

The first algorithm we describe is taken from [16] and achieves the optimal resource allocation in a max-min fairness sense when each user in the network is associated exactly to one channel, and vice-versa, a channel serves exactly one user. Therefore, $N_u = N_{ch}$.

A convenient way to model the channel allocation problem is through a matching on a bipartite graph. Let the vertices $i = 1, 2, \dots, N_u$ represent the users, and let the vertices $j = N_u + 1, N_u + 2, \dots, N_u + N_{ch}$ represent the channels. A solution to the allocation problem is therefore a set A of edges (i, j) such that

- $|A| = N_u = N_{ch}$
- A is a *matching*, i.e., if $(i_1, j_1) \in A$ and $(i_2, j_2) \in A$, then $i_1 \neq i_2$ and $j_1 \neq j_2$

An effective way to avoid searching the whole solution space is to sort all links $(i, j) \in A$ in order of increasing capacity that user i achieved on channel j . From this, we can proceed in steps: at each step we remove the edge with the lowest capacity and check whether this prevents from finding a feasible solution (for example because the remaining edges imply that, say, n nodes must now be allocated m channels, with $m < n$). Let (\bar{i}, \bar{j}) be the edge whose removal inhibits the solution of the problem: the capacity of (\bar{i}, \bar{j}) is then the

```

 $P_0 := P; \quad Q_0 := \emptyset; \quad k := 0;$ 
while  $|Q_k| < N$  do
  increment  $k$ 
   $(i, j) := \text{MinCapacityEdge}(P_{k-1})$ 
   $P_k := P_{k-1} \setminus \{(i, j)\}$ 
   $M_k := \text{HighestCardinalityMatching}(P_k)$ 
  if  $|M_k| + |Q_{k-1}| < N$  then
     $Q_k := Q_{k-1} \cup (i, j)$ 
    for all  $m$  such that  $(i, m) \in P_k$  do
       $P_k := P_k \setminus \{(i, m)\}$ 
    end for
    for all  $n$  such that  $(n, j) \in P_k$  do
       $P_k := P_k \setminus \{(n, j)\}$ 
    end for
  else
     $Q_k := Q_{k-1}$ 
  end if
end while
return  $Q_k$ 

```

Figure 2. CFDMA algorithm for max-min fair channel allocation [16].

maximum, over all feasible allocations, of the minima of the channel capacities in each allocation; thus, the value of the capacity of (\bar{i}, \bar{j}) will appear in the capacity vector of the max-min fair channel allocation. After having identified the link (\bar{i}, \bar{j}) as belonging to the solution, we remove all other edges starting from user \bar{i} or incident on channel \bar{j} , which have already been assigned to each other, and continue by solving the same problem for the allocation of $N_u - 1$ users onto $N_{ch} - 1$ channels. The algorithm terminates when all users and channels have been allocated.

To determine whether a feasible solution exists, note that each time we remove an edge, the remaining edges must guarantee the existence of a matching of cardinality at least equal to the number of users that have still to be allocated. We have to perform this check at each removal: the first time the check fails, we know that the last removed edge belongs to the max-min fair solution. In order to perform the check, it suffices to find a highest-cardinality matching (HCM) over P , which is defined as the set of all edges that can belong to a solution, i.e., $P = \{(i, j) \mid i = 1, \dots, N_u, j = N_u + 1, \dots, N_u + N_{ch}\}$. If the cardinality of the HCM is less than the number of users (and channels) to be allocated, then the assignment is not complete and the last removed edge belongs to the solution.

A precise formulation of the algorithm is provided in Figure 2 [16], where the `MinCapacityEdge()` operation simply returns the edge which has minimum capacity in the given set. The algorithm is dubbed Cognitive FDMA (CFDMA) because it allocates resources based on the knowledge of a quality metric related to user-channel pairs, represented by link capacity in this case.

B. Resource allocation in the case $N_{ch} = kN_u$

The algorithm presented in section III-A provides optimal allocation in case $N_{ch} = N_u$. However, a more efficient resource allocation and balance between the performance of all users might be easier to achieve if the number of users to allocate is less than the number of available channels. The reason behind this fact lies in the greater number of

degrees of freedom that the algorithm can exploit: namely, the finer granularity of the spectrum subdivision allows allocation choices to pinpoint the very channels where a user experiences the best performance. Unfortunately, the problem of allocating N_u users onto kN_u channels in order to maximize some fairness metric is known to be NP-hard [12], and thus needs to be approached through suboptimal heuristic methods. Our aim in the following is to propose efficient heuristics that achieve a good solution to the allocation problem, without requiring excessive computational complexity in order to be evaluated.

For example, assume that $N_{ch} = kN_u$, with $k > 1$. The CFDMA algorithm in Figure 2 can be applied to this case as well, even if, for simplicity, it has been described for the case $k = 1$ in subsection III-A. A straightforward way to do this is to perform a first run, in order to allocate exactly N_u channels, one per user. These channels (as well as all edges connected to them) are then removed from the graph. A second run of the CFDMA algorithm will result in the allocation of N_u more channels, one more per user, and to their removal from the graph. Therefore, it suffices to run the CFDMA algorithm exactly k times to allocate exactly k channels per user (hereafter, “ k CFDMA”). The result obtained this way, however, is suboptimal in that the max-min fair allocation is searched for “locally,” i.e., at each of the k times the CFDMA algorithm is run. In order to improve the quality of the solution found, better heuristics need be applied: one of these (the one we found to be more effective) is described in the next subsection.

C. The Usage-Value-based allocation

The following allocation technique is adapted from [18], where it has been proposed to improve the throughput of an Orthogonal Frequency Division Multiplexing (OFDM) system. The algorithm works in two phases. During the first *initialization phase*, all channels are assigned to the users according to some prescribed criterion. This first allocation is required to fix a starting point from which to iterate toward the final solution. For example, a way to assign channels is to give exactly k channels to each user, by choosing those where that user experiences the best conditions (this method is dubbed “even” in the following). A second way is to assign channels proportionally to the ratio of the user’s maximum capacity (taken on all channels) to the sum of all maximum capacities as experienced by all users. (“proportional” in the following). Regardless of the initial assignment, the algorithm proceeds by having each user calculate its own Usage Values (UVs) for every channel $j = 1, \dots, N_{ch}$. These values are defined in this phase as the capacity experienced by the user on any channel j . For each user, a higher UV related to a certain channel implies a greater probability that the user actually selects that channel. At this point, each user weighs its usage values so as to increase the opportunity that the best channels are in fact chosen. To do this, the greatest UV seen by a user is multiplied by N_{ch} , the second greatest UV by $N_{ch} - 1$, and so forth. Finally, all weighed UVs are normalized to 1. This is repeated for all users.

The first phase is completed by having every user choose the k channels where it experiences the highest UVs. Notice

that, up to this point, no constraint has been set to the number of users that may be assigned a channel. The objective of the second phase of the UV algorithm is in fact to settle any “collision” of more than one users onto the same resource, as well as to make sure that no channel is left unused.

To this end, the second phase (the *iteration phase*) updates the UVs so as to make it less likely to choose those channel where many users collided during the first phase. To this end, let $U_c(t-1)$ be the usage value of a given channel c at iteration $t-1$, and call $C_c(t-1)$ the *cost* of channel c , defined as the number of *other* users that have collided on the channel. Every user modifies its own UVs according to the following first-order relationship [18]:

$$U_c(t) = wU_c(t-1) + \frac{U_c(t-1)}{\frac{C_c(t-1)}{N_u-1} + 1}, \quad (4)$$

which has the net effect of reducing the UV of any channel which has been selected by more than one user, thereby making its choice less likely. The weight w in (4) balances the stability and convergence speed of the algorithm. Simulation results show that a good value lies in the interval $[0.4, 0.6]$: for this reason, we will assume $w = 0.5$ in what follows. After the update of the UVs, each user chooses again its best k channels; this procedure is performed iteratively until no channel is chosen by more than one user. In case the algorithm exceeds a given number of iterations, any such channel is assigned according to heuristic criteria (for example, by allocating the channel to the contending user with lowest overall capacity).

D. Extensions of the UV allocation scheme

Depending on the size of the network area (i.e., on the maximum distance from any user to the common sink), the UV algorithm may be more or less effective. The main issue with UV is that, regardless of the initial proportional or even channel assignment, all farther users tend to have higher UVs on low-frequency channels and, as such, tend to choose the same set of channels, which will then have to be split among them. While the UV processing in (4) slightly relieves this problem, in larger networks, the overcrowding of the low-frequency portion of the acoustic spectrum is more difficult to handle, especially if many users are present.

To partially solve this issue, we have considered two methods to heuristically redistribute the channels among the users after the UV algorithm has converged:

- *Realloc1*: the user with *largest* overall capacity *transfers* its lowest-frequency channel to the user with smallest capacity;
- *Realloc2*: the user with *lowest* overall capacity *acquires* the channel where it would experience the best performance, among those owned by other users.

In either case, the process is driven by the same considerations that were behind allocation 2 in Figure 1, i.e., that far users require low-frequency channels to transmit efficiently. In order to avoid that either method iterates indefinitely, proper stopping conditions are set. For example, in *Realloc1*, channel donors stop transferring channels if their own capacity becomes the smallest among all users; furthermore, no node can be asked

to donate more than $n_c - 1$ channels, if n_c is the number of channel that user was initially assigned. Similar criteria apply to method *Realloc2* as well.

E. Complexity considerations

It is important to understand the computational efforts required by each method, as a further measure of the algorithms’ feasibility. CFDMA requires to check the graph edge with lowest capacity at every step, which can be done with constant complexity if the edge capacities have been sorted initially (complexity $\mathcal{O}(N_{ch}^2 \log N_{ch})$). The `HighestCardinalityMatching()` operation can be solved efficiently using well-known graph theory techniques such as the augmenting path algorithm and the Hopcroft-Karp algorithm [19], with complexity $\mathcal{O}(N_{ch}^2 \sqrt{N_{ch}})$ in the latter case. The number of steps performed by the max-min capacity channel allocation algorithm is bounded by the number of edges, which is N_{ch}^2 . Therefore, the overall complexity of our algorithm is $\mathcal{O}(N_{ch}^{9/2})$, and the cost of initially sorting the set of edges is negligible. The complexity of the k CFDMA algorithm is hence straightforwardly derived as $\mathcal{O}(kN_{ch}^{9/2})$, where $k = N_{ch}/N_u$ represents the number of times the CFDMA algorithm is run.

For the UV algorithm $\mathcal{O}(N_u N_{ch} \log N_{ch})$ is the cost of initially sorting, weighing and normalizing all UVs at all users by the end of the first phase, whereas each iteration of the second phase requires $\mathcal{O}(N_u N_{ch})$ operations to update the UV values and $\mathcal{O}(N_u N_{ch} \log N_{ch})$ to sort them again, where we recall that $N_u = N_{ch}/k$. Therefore, the complexity of the UV algorithm is $\mathcal{O}(N_{ch}^2 \log N_{ch})$, which is indeed much lower than required by the CFDMA algorithm. The extensions considered in subsection III-D would add a further worst-case complexity of $\mathcal{O}(N_{ch}^2 \log N_{ch})$ to sort the channel capacities, whereas the cost of selecting the best channel to acquire (or to transfer to another user) is negligible after sorting. Hence, channel redistribution methods do not significantly worsen the complexity of the UV approach.

IV. RESULTS

In this section, we present relevant results that assess the performance of our heuristic channel allocation schemes as compared to a plain FDMA scheme (i.e., where the channel assignment is oblivious of distance, and thus of capacity), and to the CFDMA scheme, which is optimal in a max-min fairness sense, in case $N_{ch} = N_u$. The comparison is carried out by means of simulation, using the NS-Miracle framework [20] together with a module [21] which implements the underwater channel model [15] summarized in Section II. We considered a scenario where the network nodes must transmit data to a common sink. The sink is placed in the middle of a circular area of given radius, equal to the coverage range of the sink; a varying number of users is randomly placed in this area. All users transmit using a channel allocated in the 10 kHz–40 kHz band. For FDMA and CFDMA, this band has been subdivided into as many equal-bandwidth channels as the number of users, whereas for the heuristic allocation algorithms we created $N_{ch} = kN_u$ channels. Thorough simulations have shown that

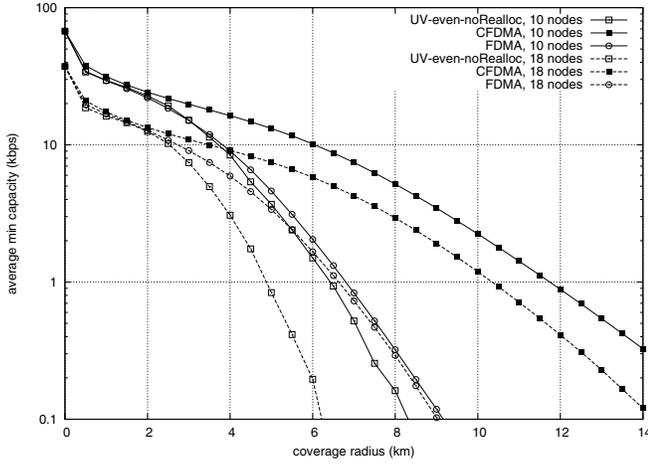


Figure 3. Average minimum capacity as a function of the network size, for networks with 10 and 18 nodes. All policies, $k = 1$.

choosing $k > 3$ does not yield appreciably better performance than $k = 3$, which is hence the configuration we assume in the following, unless differently specified.

For the purpose of this evaluation, all users transmit signals with a constant power spectral density of 97 dB re μPa per Hz in the allocated channels, and zero outside those channels.² The channel allocation is performed by a centralized controller at the beginning of the simulations, and is carried out randomly for the FDMA case, or using one of the channel-aware algorithms described in section III otherwise.³ We run our simulations for different values of the number of users, N_u , and of the coverage radius of the sink. All performance figures are averaged over more than 1000 independent random user placements, as needed to achieve 95% confidence intervals that are small compared to the sample average values.

Our first comparison is carried out among the minimum capacity achieved among all network users by using the FDMA, CFDMA and UV channel allocation techniques, and is presented in Figure 3 for the simpler case $N_u = N_{ch}$, i.e., $k = 1$. This figure supports the fact that the CFDMA approach is optimal in a max-min fair sense, if $k = 1$. We consider networks with 10 or 18 nodes, in order to model less or more constrained situations, respectively. From Figure 3, we see that the UV allocation algorithm in this case yields even worse performance than plain FDMA, for a network radius of more than 5 km: one of the reasons behind this fact is that, especially in larger networks, a high number of users measure large UVs in the low-frequency spectrum region, and thus concentrate their preference on low-frequency channels. Since each user can be assigned at most one channel in this case, some far nodes will have to move to medium-frequency channels, where they will experience much worse performance. This was among the main arguments

²We note that accounting for a more realistic signal psd and transmission mask is possible and would not change qualitatively our results.

³Note that, in a real scenario, the controller would require a means of gathering distance information among the sink and its neighbors. This can be implemented in underwater networks with good accuracy, by measuring the round-trip time of acoustic pulses.

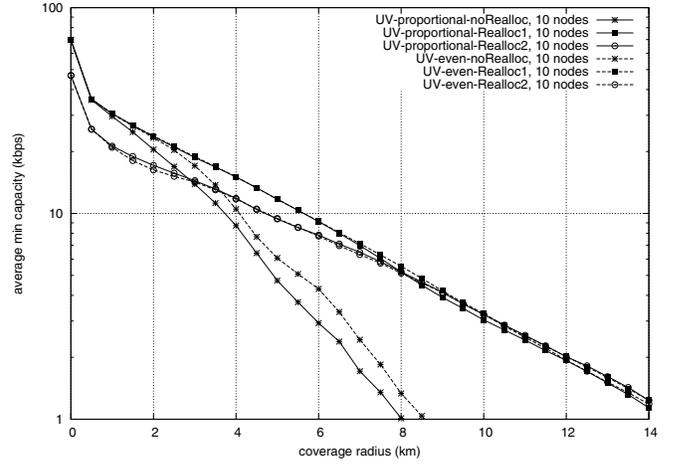


Figure 4. Average minimum capacity for all variants of the UV method as a function of the network size, 10 nodes, $k = 3$.

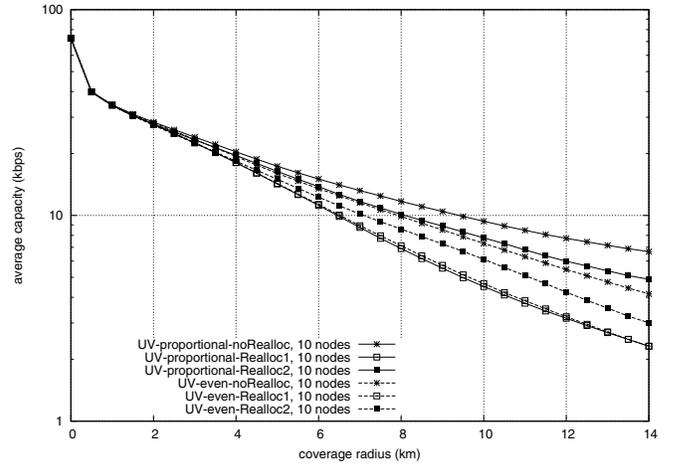


Figure 5. Average capacity achieved by different versions of the UV algorithm as a function of the network size, 10 nodes, $k = 3$.

behind the choice of performing post-assignment heuristic channel re-allocation after the UV algorithm has converged. Figure 4 shows a comparison between the minimum capacity performance of all variants of the UV algorithm, with even or proportional initial channel share (see section III-C), and using Realloc1, Realloc2 or no re-allocation method at all. We set $k = 3$ here. We observe that using Realloc1 or no re-allocation is best for small networks. After the network size increases beyond 3 km, the minimum capacity decreases considerably if no re-allocation is used. Nevertheless, after 7 km, Realloc1 and Realloc2 achieve almost the same performance.

Some further details about the performance of UV methods can be seen in Figure 5, which shows the average capacity, taken over all network users, that is achieved by each version in the same network setting introduced before. This figure shows that the loss in the channel capacity of the worst user corresponds to a gain in the average capacity over the whole network. More specifically, with both the proportional and the even initial channel assignments, using no re-allocation achieves the best average capacity. Conversely, Realloc1, which has already been shown to provide the best minimum

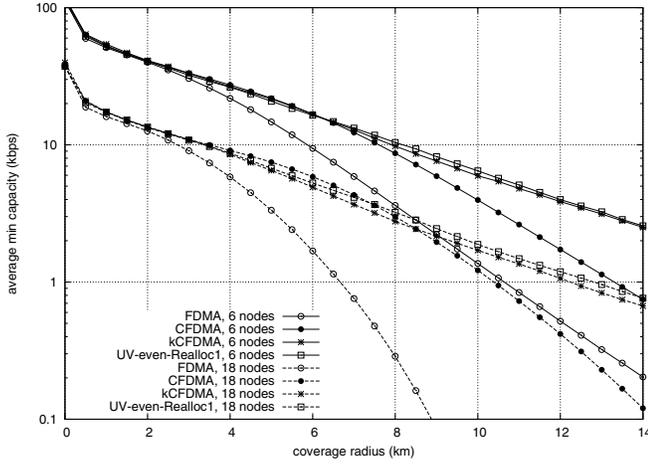


Figure 6. Average minimum capacity as a function of the sink coverage radius for all policies, $k = 3$.

capacity, achieves the lowest average performance. After these results, and recalling that our objective here is to guarantee the best minimum performance, we choose to employ only UV with Realloc1 in the following.

Figure 6 summarizes the above evaluations by showing a performance comparison of all policies for $k = 3$, with small or larger networks (6 and 18 nodes, respectively), considering both the k CFDMA and the UV algorithms. For completeness, we have plotted two more sets of curves in this graph, which correspond to the FDMA and CFDMA algorithms (for which $k = 1$). Figure 6 shows the effectiveness of the UV allocation scheme with even initial channel share and the Realloc1 post-processing: in both the 6 and the 18 users case, this algorithm offers the best performance, showing the smoother decay of the minimum capacity with the network size. Interestingly, the k CFDMA algorithm achieves slightly worse performance, and in addition requires greater computational complexity, see subsection III-E. Therefore, the UV algorithm yields a two-fold advantage. As expected, FDMA achieves the worst minimum capacity instead, whereas CFDMA keeps a good performance level until the network becomes too large (8 km size or more), after which the UV and k CFDMA approaches can profitably exploit the greater number of degrees of freedom granted by having kN_u channels to allocate to N_u users.

Figures 7 and 8 provide deeper insight, by showing the cumulative distribution functions (cdf) of the capacity, i.e., the probability that the actual capacity experienced by the users after the allocation has been performed is less than or equal to the value indicated in the abscissa. We have considered small to medium-size networks, with either 6 (Figure 7) or 18 nodes (Figure 8). These results show that the UV technique is also the most fair, besides being the best in terms of minimum capacity. Consider, for example, the case of a 6 km network with 18 nodes (leftmost curves in Figure 8). The UV approach grants a capacity of at least 1 kbps to any node, and moreover its cdf shows the fastest increase from 0 to 1, which denotes a strong concentration of most capacity values around 1 kbps. Conversely, FDMA yields the worst performance, as confirmed by the higher left tail of its cdf, which shows

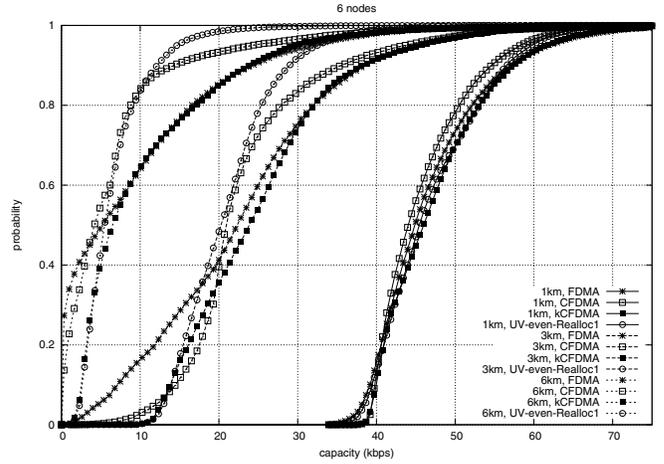


Figure 7. Cumulative distribution functions of the capacity for a 1, 3 and 6 km network size, 6 nodes.

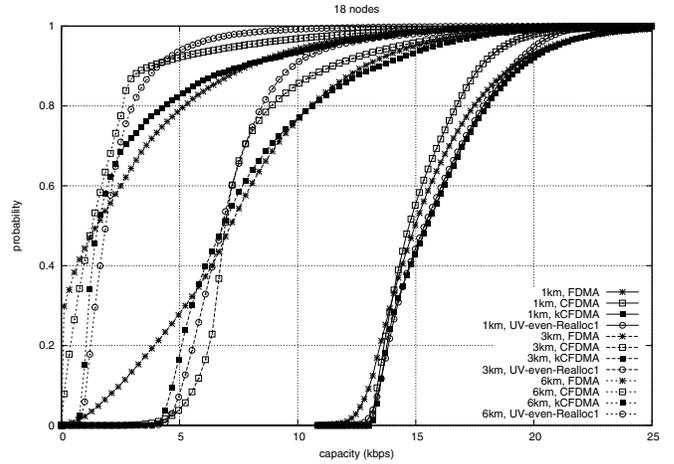


Figure 8. Cumulative distribution functions of the capacity for a 1, 3 and 6 km network size, 18 nodes.

a considerably high probability that a capacity of less than 0.1 kbps is achieved at the worst user. Similar considerations apply to the other sets of curves (for different number of nodes and network size).

We conclude this study by comparing the throughput performance of our deterministic frequency-division access schemes with a plain ALOHA protocol, whereby all nodes transmit on the whole system bandwidth regardless of distance. In this case, due to the longer transmit time required by farther nodes and to the uncontrolled simultaneous access, the system suffers from collisions, that lead to the loss of all packets involved. With deterministic frequency-division access, instead, the allocation of separate channels avoids any collision between different users, at the price of a lower per-user transmit rate. We considered the UV scheme with even initial share and Realloc1 ($k = 3$) as well as the CFDMA scheme ($k = 1$). Figure 9 details this comparison against the packet generation rate per node (λ packets per second) for a network of 6 nodes and 1 km radius, a scenario where the CFDMA and UV schemes tend to have comparable performance (see Figure 6). Our results show that ALOHA has a predictably

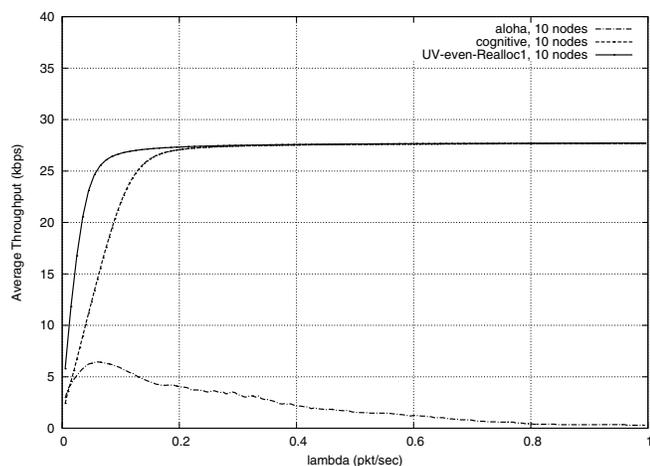


Figure 9. Comparison between ALOHA and deterministic channel allocation-based algorithms. Both UV ($k = 3$) and CFDMA ($k = 1$) are considered. The network is made of 10 nodes distributed over a 1 km range.

low performance, even though it outperforms deterministic access for very low traffic generation rates. Conversely, both CFDMA and UV support more traffic and converge to a stable performance at high λ . Moreover, UV yields a further advantage, in that its throughput is greater for lower λ and converges more steeply to the maximum traffic sustainable by the scheme. This is a consequence of the good management of the degrees of freedom allowed by the allocation of N_u users onto kN_u channels, which once again confirms the goodness of this research direction.

V. CONCLUSIONS

In this paper, we approached the problem of channel allocation in the frequency domain for underwater networks. We proposed and evaluated efficient assignment techniques, with the objective to come up with a max-min fair solution, or a good heuristic approximation of it, especially in the more complex case when there are more channels to assign than users. Our results show that considerable advantages can be gained in terms of both minimum capacity and fairness, even with heuristic methods, with a proper choice of the assignment options.

Future directions of this work include the definition of a completely distributed MAC protocol based on the channel assignment ideas deployed here, as well as the evaluation of the same ideas using realistic modulation formats.

REFERENCES

- [1] G. Maral and M. Bousquet, *Satellite communications systems, techniques and technology*. John Wiley and sons, 2003.
- [2] J. G. Proakis, *Digital Communications*, 3rd ed. McGraw-Hill, 1995.
- [3] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," *IEEE J. Oceanic Eng.*, vol. 25, no. 1, pp. 72–83, Jan. 2000.
- [4] P. Casari, S. Marella, and M. Zorzi, "A comparison of multiple access techniques in clustered underwater acoustic networks," in *Proc. IEEE/OES Oceans*, Aberdeen, Scotland, June 2007.
- [5] Jose dos Santos Coelho, "Underwater acoustic networks: evaluation of the impact of media access control on latency in a delay constrained network," Master's thesis, Naval Postgraduate School, Monterey, CA, Mar. 2005.
- [6] L. G. Roberts, "ALOHA packet system with and without slots and capture," *ACM SigComm Computer Communication Review*, vol. 5, no. 2, pp. 28–42, 1975.
- [7] M. Molins and M. Stojanovic, "Slotted FAMA: a MAC Protocol for underwater acoustic networks," in *Proc. MTS/IEEE Oceans*, Singapore, Sept. 2006.
- [8] M. K. Park and V. Rodoplu, "UWAN-MAC: an energy-efficient MAC protocol for underwater acoustic wireless sensor networks," *IEEE J. Oceanic Eng.*, vol. 32, pp. 710–720, July 2007.
- [9] B. Peleato and M. Stojanovic, "A MAC protocol for ad hoc underwater acoustic sensor networks," in *Proc. ACM WUWNet*, Los Angeles, CA, Sept. 2006, pp. 113–115.
- [10] P. Casari, F. E. Lapicciarella, and M. Zorzi, "A detailed simulation study of the UWAN-MAC protocol for underwater acoustic networks," in *Proc. MTS/IEEE Oceans*, Vancouver, Canada, 2007.
- [11] J. Rice *et al.*, "Evolution of seabed underwater acoustic networking," in *Proc. MTS/IEEE Oceans*, 2000, pp. 2007–2017.
- [12] M. R. Garey and D. Johnson, *Computers and intractability: a guide to the theory of NP-completeness*. W. H. Freeman, 1990.
- [13] R. Urick, *Principles of Underwater Sound*. McGraw-Hill, 1983.
- [14] L. Berkhovskikh and Y. Lysanov, *Fundamentals of Ocean Acoustics*. Springer, 1982.
- [15] M. Stojanovic, "On the relationship between capacity and distance in an underwater acoustic communication channel," *ACM Mobile Comput. and Commun. Review*, vol. 11, no. 4, pp. 34–43, Oct. 2007.
- [16] N. Baldo, P. Casari, and M. Zorzi, "Cognitive spectrum access for underwater acoustic communications," in *Proc. IEEE ICC CogNet Workshop*, Beijing, China, 2008.
- [17] D. Bertsekas and R. Gallager, *Data Networks*. Prentice Hall, 1992.
- [18] T. C. H. Alen, A. S. Madhukumar, and F. Chin, "Capacity enhancement of a multi-user OFDM system using dynamic frequency allocation," *IEEE/ACM Trans. Networking*, vol. 49, no. 4, pp. 344–353, Dec. 2003.
- [19] J. E. Hopcroft and R. M. Karp, "An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs," *SIAM Journal on Computing*, vol. 2, no. 4, pp. 225–231, 1973.
- [20] N. Baldo, F. Maguolo, M. Miozzo, M. Rossi, and M. Zorzi, "Ns2-miracle: a modular framework for multi-technology and cross-layer support in network simulator 2," in *ACM NSTools*, Nantes, France, October 2007.
- [21] "Model for underwater channel in ns2," 2007. [Online]. Available: http://www.dei.unipd.it/research/signet/tools/ns_underwater