

A scalable analytical framework for deriving optimum scheduling and routing in underwater sensor networks

Francesco Lo Presti
Dipartimento di Informatica, Sistemi e Produzione
Università di Roma Tor Vergata
lopresti@info.uniroma2.it

Chiara Petrioli, Roberto Petroccia and Ariona Shashaj
Dipartimento di Informatica
Università di Roma “La Sapienza”
{petrioli,petroccia,shashaj}@di.uniroma1.it

Abstract—Underwater sensor networks have become an important area of research with many potential practical applications. Given impairments of optical and radio propagation, acoustic communication is used for underwater networking, which translates into variable and long propagation delays, low data rates, long interference ranges and significant fluctuations in terms of link quality over time. A complete characterization of the unique features of the acoustic channel introduces significant complexity both in analytical models and in simulators but is needed for correct characterization of underwater protocols performance. Our objective has been that of designing scalable analytical techniques which are able to derive optimum traffic scheduling and routing for underwater sensor networks while accurately capturing underwater channels features. Specifically the paper presents an analytical model for joint MAC and routing optimization which produces the optimum solution for small to medium scale underwater networks. Scalable, centralized heuristics are then designed, which combine approximate analytical models and scheduling heuristics, and are able to generate solutions close to the optimum. The overall result is a powerful tool to derive benchmark results (upper bounds) for underwater protocol performance and to understand the trade-offs and performance limits of such systems.

I. INTRODUCTION

Underwater wireless networking has been recognized as an enabling technology with potential practical impact on a host of different collaborative applications, such as environmental monitoring, underwater safe CO₂ storage, coastline protection, early warning systems for seaquakes and earthquakes, assisted navigation and exploration of oil-well [1]–[5], just to cite a few. Given impairments of optical and radio propagation underwater, the typical physical layer technology for underwater wireless sensor networks is acoustic communication, which translates into variable and long propagation delays, low bandwidth and data rates, sound speed variability, long interference ranges and significant fluctuations in terms of link quality over time. A complete characterization of the unique features of the acoustic channel introduces significant complexity both in analytical models and in simulators but is needed for correct characterization of underwater protocols performance. Similar to terrestrial sensors, underwater sensors are powered by batteries, have limited resources and forward

data in a hop-by-hop fashion. Replacing and recharging batteries is very difficult, so prolonging the network lifetime and maximizing the communication capabilities are important aspects in Underwater Wireless Sensor Networks (UWSNs).

In this paper we study the joint optimal routing/scheduling problem for underwater sensor networks with periodic traffic. Joint routing/scheduling solutions for wireless networks have been widely investigated in the literature. In general, determining a feasible routing scheme together with an overall transmission schedule is an interesting and challenging problem [6], [7]. In underwater networks the problem is exacerbated because of the need to explicitly model propagation delay and interference constraints. When studying terrestrial radio networks the link propagation delay is usually assumed to be zero. This assumption cannot be made when modeling underwater acoustic networks where the propagation speed of acoustic signals (around 1500 m/s) is roughly five orders of magnitude slower than that of RF signals. This is usually called “Spatial-Temporal Uncertainty” [8]. Moreover, for underwater acoustic communication the signal power attenuation is lower than the one for terrestrial radio networks, which produces an interference range potentially much longer than the transmission one [9], [10]. Therefore, considering all the differences between RF communication in air and acoustic transmission underwater, solutions for terrestrial wireless (sensor) networks cannot be directly applied to UWSNs. Moreover, assumptions underlying analytical models for WSNs make such models not extendable to correctly and accurately model performance in UWSNs.

In this paper we propose a new optimization framework for underwater sensor networks. Specifically, we provide the following contributions:

- We formulate the routing/scheduling problem into an Integer Linear Programming (ILP) model which yields the joint optimal routing and scheduling which maximizes the network throughput and minimizes the energy consumption. Differently from most of prior research in the area, we consider an accurate interference model where we account not only for the presence of other single transmitters but also for the combination of multiple

transmissions that can overlap during the reception at the packet destination. The resulting solution can only scale to a few tens of nodes. This is in any case larger than current deployments but demands for more scalable heuristic solutions to understand the performance trade-offs of future UWSNs.

- We have therefore proposed a heuristic solution which scales well with the problem size. Since the complexity of the optimal strategy stems from the need to jointly optimize routing and scheduling, in the heuristic we use the “divide et impera” principle: First we compute the routing; then, with the routing fixed, we schedule packet transmissions. Our scheduling heuristic is motivated from the observation that the scheduling problem can be regarded as a specialized vertex-coloring problem of the network conflict graph which also accounts for the large propagation delays.
- We have performed an extensive evaluation of the proposed solutions. Several variants of the heuristic have been analyzed by means of simulations, in different possible UWSN deployment scenarios, and compared (for the small and medium scale ones) with the results of the analytical model. Both the analytical model and simulations use the Bellhop ray tracing model [11] as transmission loss and propagation models, providing a more accurate description of the underwater acoustic channel behavior with respect to classical empirical formulas derived from [12].

Our experiments show that, for network sizes where the optimum can be derived, the proposed heuristic produces results very close to the optimal ones (we observed a throughput degradation of 7% on average) at a fraction of time. Moreover, the proposed heuristic is able to scale to larger networks. The proposed solution can therefore be used as guideline and benchmark for the design and evaluation of distributed protocol stacks.

The remainder of the section is organized as follows. In Section II we discuss related results in the literature. In Section III we present our system model. In Section IV we formulate an ILP model for joint routing-scheduling optimization. Section V presents our scalable centralized heuristic solution for traffic scheduling. In Section VI we present the results of our comparative experimental evaluation of the proposed solutions. Concluding remarks are given in Section VII.

II. RELATED WORKS

In this section we focus on papers which have addressed optimization and analytical models for UWSNs. In [13] an optimal traffic scheduling solution, named ST-MAC, is formulated using a weighted, directed conflict graph with the aim to minimize the frame size while achieving the maximal throughput for one-hop acoustic networks. ST-MAC schedules transmissions by assigning a color (an integer) to each edge in the conflict graph. Given the problem complexity, the authors propose an approximate algorithm based on a greedy heuristic of the vertex coloring problem. In [14] the scheduling

problem is reduced to a standard TSP so that existing heuristics can be directly applied. Scheduled transmissions can start at any time, which improves channel utilization as well as network throughput. Both these solutions address only one-hop acoustic networks, and consider only pairwise interference among links. As our experiments show, in UWSNs, due to the large interference range, a more accurate interference model should be considered, which accounts also for interference caused by multiple transmissions which overlap destructively during packet reception.

A joint optimization of placement, scheduling and routing is presented in [15] with the goal to minimize energy consumption, modeling the behavior of the network during a large single frame. The paper investigates small-scale multi-hop networks considering the presence of multiple interfering nodes and the use of an underwater acoustic channel attenuation model, which is based on empirical formulas derived from [12]. Although this work considers a more accurate model than previous results, the model formulation is quite complex and can be applied only to small-scale networks. Moreover, a more realistic and accurate underwater acoustic attenuation model should be used with respect to empirical formulas. To be able to predict UWSN performance in realistic settings the model should also be able to generate the steady state routing and scheduling, rather than simply follow the system evolution for some slots. In our work, similarly to [15], we jointly address the routing/scheduling problem. We focus on multi-hop networks, considering an accurate interference model and assuming as channel attenuation model the Bellhop ray tracer [11]. Bellhop is used with historical environmental data and provides us with a more accurate description of the underwater acoustic channel behavior with respect to empirical formulas.

III. SYSTEM MODEL

1) *Network Model:* We model the network with a directed graph $G(V, E)$, where V is the set of sensor nodes and E is the set of links available in the network. Each node is located in a three-dimensional space modeling the underwater deployment area and transmits at a fixed power level P . There is a link from node u to v if a reliable transmission can occur between the two nodes, i.e. if the *Signal-to-Noise Ratio* (SNR) of u 's transmission at v is higher than a given threshold SNR_{th} . Given the carrier frequency f , for each pair of nodes (u, v) we compute the transmission gain $Gain(u, v)$. Then $(u, v) \in E$ if and only if $\frac{P \cdot Gain(u, v)}{N} \geq SNR_{th}$, where $N = N(f) \cdot \Delta f$ is the ambient noise power. $N(f)$ is the noise power spectral density as described in [16]; Δf is the receiver noise bandwidth (a narrow band around the frequency f). The propagation delay $d(u, v)$ between u and v is expressed in number of slots and it is obtained by the link length divided by the product of sound speed (we assume a sound speed in water of 1500m/s) and the duration of the nominal time slot in seconds (slot_duration). For each node v , we also denote with $E^{out}(v)$ the set of outgoing links from v , and with $E^{in}(v)$ the set of links entering v .

2) *Traffic Model*: We consider a scenario where a sink node is in charge of collecting all the information generated from the sensors. We will denote by $g(v, t)$ the number of packets generated by node v at time t . Packets which cannot be transmitted directly to the sink are relayed through intermediate nodes. Based on the fact that sensor traffic for underwater monitoring is expected to be highly periodic, we consider a periodic scheduling of transmissions from the nodes. The fundamental period, called frame, is divided into multiple slots each of a fixed length, representing the time needed to transmit a data packet which we assume equal for all nodes. While a data packet has to be transmitted at the beginning of a slot, receptions can happen at any time according to transmitter/receiver propagation delay.

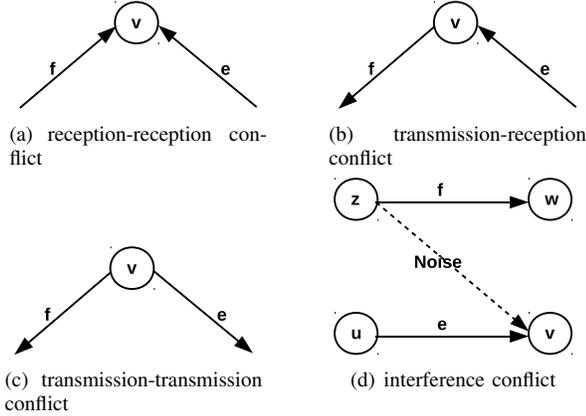


Figure 1. Conflicts.

3) *Conflicts and Interference Model*: Link conflicts/interference reduce the number of transmission opportunities nodes have to transmit packets. Conflicts occur simply because a node cannot receive and transmit at the same time or because a node cannot receive from two distinct transmitters at the same time. Interference conflicts occur when the transmission over a link prevents correct reception over another link. In the literature, most of the optimality frameworks for scheduling/routing are based on simplified interference assumptions, where necessary and sufficient conditions for the correct delivery of a packet are that this is the only transmission which takes place in the transmitters and receivers radio ranges. Basically, this means that both hidden and exposed terminal problems must be avoided. This leads to developing nice and clean mathematical constraints, which are however not suitable for the underwater scenario, given that destructive interference is typically due to multiple far away transmissions [10].

Differently from these approaches, we consider a *complete* interference model which takes into account not only the presence of other single transmitters but also the combination of multiple transmissions that can overlap during the reception at the packet destination. We want to avoid that the received Signal-to-Interference Ratio (SIR) is too low for some received packets, e.g., since many far-away nodes are all transmitting (simultaneously or at different times) with the result that the overlapping receptions at some receiver impair correct

reception.

We classify conflicts in two general categories:

a) *duplex conflict*: Since we assumed half-duplex communication, a node cannot receive packets from more than one link at time (Figure 1(a)), and it cannot transmit simultaneously on more than one link (Figure 1(c)). Moreover, it cannot transmit while it has not completely received a packet (Figure 1(b)).

b) *interference conflict*: This conflict occurs when a concurrent transmission on link f may disturb the reception on link e (Figure 1(d)). If the interference generated by transmission on f is too powerful, the packet transmitted on link e cannot be correctly received.

Let us denote with $I(e)$ the set of possible interfering links with link e .

$$\forall f \in E, f \in I(e) \Leftrightarrow \frac{P \cdot \text{Gain}(e.\text{src}, e.\text{dst})}{P \cdot \text{Gain}(f.\text{src}, e.\text{dst}) + N} < \text{SIR}_{th}$$

where $\text{Gain}(e.\text{src}, e.\text{dst})$ is the gain of signal transmitted on link e , $\text{Gain}(f.\text{src}, e.\text{dst})$ is the gain of signal transmitted by the source of link f and heard by destination of link e , and P is the transmission power that we assume equal for all nodes. The expression can be easily generalized to the case of multiple interferer links (Equation 8).

IV. OPTIMIZATION MODEL

Given the network model $G(V, E)$, we consider periodic sensors traffic generation. Our goal is to determine the minimum frame length T (thus maximizing network throughput) and the associated transmission schedule - which avoids conflicts among link transmissions - while minimizing the overall energy consumption. A schedule can be specified as a sequence S_t , $t = 0, 1, \dots, T - 1$, of sets of links, that may transmit simultaneously without conflicts. In solving the problem we exploit the periodic network behavior, and study a solution where the packet generation scheme, routing and scheduling are repeated through frames. Under these assumptions an optimal solution for a single frame, necessarily results in an optimal solution over the lifetime of the network.

We found that the joint determination of the frame length T and the associated periodic schedule is an extremely difficult problem, complicated by the fact that T should be both a variable and a parameter to be accounted for when writing the conflicts/interference constraints (see below). To overcome this difficulty we tackle the computation of the frame length T and the associated periodic transmission schedule iteratively. The idea is simply to solve the joint periodic routing schedule problem for a fixed frame length T and then to carry out a binary search to determine the optimal schedule length T^* .

Let $\text{OPT}(T)$ denote the optimal joint periodic routing schedule problem for a fixed frame length T and let $S = S(T)$ denote the associated optimal schedule (under the assumption that $\text{OPT}(T)$ has a feasible solution). Then the overall optimal schedule S^* is $S^* = S(T^*)$ where $T^* = \arg \min_T S(T)$ is the minimum frame length for which a feasible solution exists. $\text{OPT}(T)$ can be formulated as a centralized ILP problem as follows.

Table I
NOTATION SUMMARY

v, u	nodes of network graph
e, f	links of network graph
$e.src, e.dst$	source node, and destination node of link e
$E^{out}(v), E^{in}(v)$	sets of outgoing/incoming links from node v
$I(e)$	set of possible interfering links with link e .
$d(u, v)$	propagation delay expressed in number of slots between node u and v
d_e	propagation delay expressed in number of slots on link e , $d_e = d(e.src, e.dst)$
$d_{f,e}$	propagation delay expressed in number of slots from the source of link f to the destination of link e , $d_{f,e} = d(f.src, e.dst)$
T	frame size
t_e, t_f	transmission time on link e , and link f
L	number of frames to look for possible transmissions conflicts
P	transmission power
N	ambient noise power
$Gain(u, v)$	gain of signal transmitted from node u to node v
SNR_{th}	Signal-to-Noise Ratio (SNR) threshold
SIR_{th}	Signal-to-Interference Ratio (SIR) threshold
$g(v, t)$	packets generated by node v at time t
$X(e, t)$	binary variable equal to 1 if link e is active during time t
$BIn(v, t)$	integer variable that counts the number of packets added to node v buffer at time t
$BOut(v, t)$	integer variable that counts the number of packets going out form node v at time t
$BSize(v, t)$	size of buffer of node v at time t

Variables

- $X(e, t)$: binary variable equal to 1 if link e is active during the time slot t ;
- $BIn(v, t)$: integer variable that counts the number of packets added to node v buffer at the beginning of time slot t ;
- $BOut(v, t)$: integer variable that counts the number of packets transmitted by node v during time t ;
- $BSize(v, t)$: size of node v buffer at time t .

The variable X represents the schedule itself. The other variables $BIn, BOut, BSize$ are used to track data flows through nodes during a frame. Table I shows a summary of the notation we used for the model formulation.

Objective Function

Our goal is to minimize the overall energy consumption. The objective function is:

$$\min \sum_{e \in E} \sum_{t=0}^{T-1} X(e, t) \cdot E_{tx} \quad (1)$$

where $E_{tx} = P \cdot \text{slot_duration}$ is the energy consumption for each transmission. Remember that we iteratively search for the minimum frame length T , which results in a feasible solution for the ILP problem, in order to maximize the network throughput.

Constraints

- *Sink constraints.* As the sink node is only a data collector, it cannot generate traffic or transmit data packets. The following

constraints are used to inhibit generation and transmission of data packets from the sink node.

Inhibition sink generation, and transmission:

$$\sum_{t=0}^{T-1} g(\text{sink}, t) = 0, \quad \sum_{e \in E^{out}(\text{sink})} \sum_{t=0}^{T-1} X(e, t) = 0 \quad (2)$$

- *Conflict Constraints.* Conflicts among link transmissions reduce the transmission opportunities. This is captured by the following sets of constraints

(i) *Concurrent transmission conflict:* These constraints prevent that a node v can transmit over two different links at the same time (Figure 1(c)). As we assume that transmissions begin at slot boundaries, we need to ensure that at most one link in $E^{out}(v)$ is active for all possible slot $t \in [0, T - 1]$. Hence:

$$\sum_{e \in E^{out}(v)} X(e, t) \leq 1 \quad \forall v \in V, \forall t \in [0, T - 1] \quad (3)$$

(ii) *Concurrent reception conflict:* These constraints prevent that a node v can receive over two different incoming links $e, f \in E^{in}(v)$ at the same time (Figure 1(a)). Assume a transmission over link e starts at time t_e and a transmissions over link f starts at time t_f . A conflict occurs whenever the reception of the two packets overlaps in time. Let $d_e = d(e.src, e.dst)$ and $d_f = d(f.src, f.dst)$ denote the propagation delay (always measured in number of slots) along links e and f . Then a conflict occurs whenever the two packet reception time intervals $[t_e + d_e, t_e + d_e + 1]$ and $[t_f + d_f, t_f + d_f + 1]$ are not disjoint (Figure 2(a)). It is easy to verify that this corresponds to the constraint

$$\forall e, f \in E^{in}(v), \forall t_e, t_f \in [0, T - 1]$$

$$X(e, t_e) + X(f, t_f) \leq 1 \quad \Leftrightarrow \quad t_e + d_e - 1 < t_f + d_f < t_e + d_e + 1. \quad (4)$$

In addition, we need to consider the case where conflicts occurs among transmissions originated in different frames (Figure 2(b)). Indeed, it is possible that $t_f + d_f > T$, since we impose no constraint on the ending of the reception of a transmitted signal within the end of the frame in which transmission started. Hence conflicts may be caused by transmission occurred in previous frames. To capture this, we move an imaginary window of length T through L consecutive frames, where L is the maximum number of frames a packet will take to reach the farthest distance within the network, i.e. it is the minimum number of frames greater or equal to the maximum propagation delay and generalize the constraint (4) as follows:

$$\forall v \in V, \forall e, f \in E^{in}(v), \forall t_e, t_f \in [0, T - 1], \forall k \in [0, L]$$

$$X(e, t_e) + X(f, t_f) \leq 1 \quad \Leftrightarrow \quad t_e + d_e - 1 < t_f + d_f - kT < t_e + d_e + 1 \quad (5)$$

where the term kT , $k \in [0, L]$, allows us to account for conflicts between transmissions occurring in different frames.

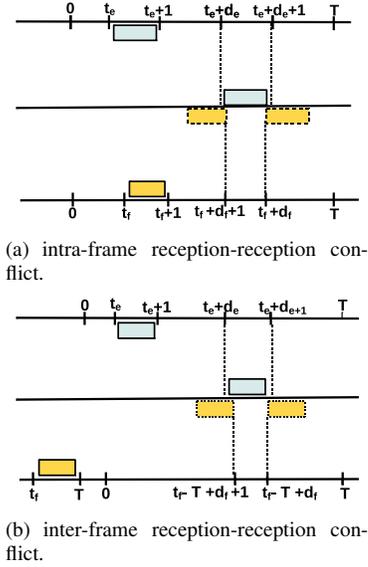


Figure 2. Concurrent reception conflicts.

(iii) *Transmission-reception conflict*: This constraint prevents that a node v can transmit a packet over link e while it is receiving data addressed to him over a different link f , where $f \in E^{in}(v)$ (Figure 1(b)). Assume node v starts transmitting over link e at time t_e and a transmission over link f starts at time t_f . A conflict occurs whenever the transmission and reception of the two packets overlap in time. Then a conflict occurs whenever the transmission interval $[t_e, t_e + 1]$ and the reception interval $[t_f + d_f, t_f + d_f + 1]$ are not disjoint (Figure 1(b)). Moreover, if $t_f + d_f > T$ we have to account for conflicts due to transmissions scheduled in previous frames. This corresponds to the constraint

$$\begin{aligned} \forall v \in V, \forall e \in E^{out}(v), f \in E^{in}(v), \\ \forall t_e, t_f \in [0, T - 1], \forall k \in [0, L] \\ X(e, t_e) + X(f, t_f) \leq 1 \quad \Leftrightarrow \\ t_f + d_f - kT < t_e < t_f + d_f - kT + 1 \end{aligned} \quad (6)$$

(iv) *Interference conflict*: This constraint prevents that while a node v is receiving over an incoming link $e \in E^{in}(v)$, correct reception is impaired due to a transmission on a different link f which is not an incoming link, i.e. $f \notin E^{in}(v)$ and $e.dst \neq f.dst$ (Figure 1(d)). Assume a transmission over link e starts at time t_e and a transmissions over link f starts at time t_f . Similarly to the concurrent reception conflict case, a conflict occurs whenever the reception of the two packets overlaps in time at v and $f \in I(e)$. Let $d_{f,e} = d(f.src, e.dst)$ denote the propagation delay (always measured in number of slots) from the source of link e to the destination of link f . Then a conflict occurs whenever the two packet reception time intervals $[t_e + d_e, t_e + d_e + 1]$ and $[t_f + d_{f,e}, t_f + d_{f,e} + 1]$ are not disjoint. It is easy to verify that this corresponds to the constraint

$$\begin{aligned} \forall v \in V, \forall e \in E^{in}(v), f \notin E^{in}(v), \\ t_e, t_f \in [0, T - 1], k \in [0, L] \text{ and } f \in I(e) \\ X(e, t_e) + X(f, t_f) \leq 1 \quad \Leftrightarrow \\ t_e + d_e - 1 < t_f - kT + d_{f,e} < t_e + d_e + 1 \end{aligned} \quad (7)$$

where again the term kT , accounts for conflicts between transmissions occurring in different frames.

We observe that this constraint captures only pairwise interference. For a more realistic interference characterization we need to consider the more general situation whereby a set of transmissions produces a non negligible interference at the receiver. This would require an exponential number of constraints which would not be feasible except for very small scenarios. Instead, we follow the approach proposed in [15] where a solution is computed at first considering only the constraints above. Then, ex-post, for the obtained schedule, we evaluate whether a SIR-violation occurs at the receiver. We have a SIR-violation at the destination of link e , if :

$$\frac{P \cdot Gain(e.src, e.dst)}{\sum_{f \notin E^{in}(v): X(f, [t-d_{f,e}])=1} P \cdot Gain(f.src, e.dst) + N} < SIR_{th} \quad (8)$$

Let A denote the set of links interfering with link e when they are all actives. If there is a SIR-violation at the destination of link e , we augment the set of constraints by adding a new one ensuring that at most $|A| - 1$ of the links in A can co-exist when link e is active. After all the violations are accounted for with the new constraints (8) the problem is solved again. We proceed iteratively until no SIR-violations are observed.

- *Buffer management*. Another aspect that is necessary to consider in the formulation of our problem is data flow management. The following constraints describe the buffer dynamic of each sensor node.

(i) *Incoming messages*: Packets that are inserted into the buffer of node v at time t are packets either generated by node v , or completely received at time t . If node v receives a packet at time t over link e with propagation delay d_e , then the source node of e started to transmit data at time $t_e = t - d_e - 1$. If $t_e < 0$, the transmission started during a previous frame. Since traffic is assumed to be periodic, if link e transmitted during the k -th previous frame at time t_e , it will transmit at time $t_e + kT$ during the current frame.

(ii) *Outgoing messages*: The total number of packets transmitted by node v at time t is the sum of the number of packets transmitted at time t over the links departing from v .

(iii) *Storage*: The number of packets inside the buffer at time t are those that arrived at node v before t , minus those transmitted by v before t .

$$\begin{aligned} \forall v \in V, t \in [0, T - 1] \\ BIn(v, t) = g(v, t) + \sum_{k=0}^L \sum_{e \in E^{in}(v): 0 \leq t_e < T} X(e, t_e) \quad (9) \\ \text{where } t_e = t + kT - d_e - 1 \\ BOut(v, t) = \sum_{e \in E^{out}(v)} X(e, t) \end{aligned} \quad (10)$$

$$BSize(v, t) = \sum_{r=0}^t BIn(v, r) - \sum_{r=0}^t BOut(v, r) \quad (11)$$

$$BSize(sink, T - 1) \geq \sum_{v \in V} \sum_{t \in [0, T - 1]} g(v, t) \quad (12)$$

Since the system dynamic is periodic with buffer sizes being equal frame after frame at each slot time, these constraints also represent the flow conservation laws. In addition, (12) ensures that all traffic generated during a frame eventually reaches the sink.

Given a solution of the problem $\text{OPT}(T)$ we obtain the schedule S_t , $t = 0, \dots, N - 1$ as follows:

$$S_t = \{e \in E | X(e, t) = 1\} \quad t = 0, \dots, T - 1 \quad (13)$$

V. CENTRALIZED HEURISTIC

In this section we present a centralized heuristic for the joint routing scheduling problem. Given the complexity of the optimal problem, we propose a simple *divide et impera* scheme where we first determine packet routing and then, with the routing fixed, we schedule packet transmissions. For routing we can, in principle, use any suitable algorithm, e.g., shortest path, geographic routing, etc. In this paper for the sake of simplicity we adopt shortest path routing with unitary link cost. This is motivated by the observation that shortest path routing, with unitary cost, minimizes the number of transmissions which is our optimal problem objective function (under the assumption of identical transmission costs).

Our scheduling heuristic is inspired by a recent work presented in [13], which proposes a greedy heuristic for scheduling in a UWSN. In [13], following the observation that the scheduling problem can be regarded as a coloring problem of the network conflict graph, where a color assigned to a node corresponds to a transmission slot in a frame, the authors presents a vertex-coloring greedy heuristic, which accounts for the interference delay and schedules links according to their loads (breaking ties in favor of links with higher number of conflicts in the network). Our proposed solution extends the work in [13] in several directions. First, we consider a full interference model: While constructing the schedule, we keep track of the interference generated by already scheduled nodes at all receivers in the networks. When scheduling a new transmission we choose available slots that avoid both conflicts between links and prevent receivers interference due to already scheduled transmissions. Second, our solution accounts for a periodic schedule by explicitly considering the interference among transmissions occurring in different frames. Finally, we consider more metrics than those considered in [13]. While their work consider only load as metric to rank links, we consider and propose alternative metrics, e.g., number of conflicting links or the ratio between the link load and available slots. In practice, given the short execution time required by the heuristic, we can compute the schedule using different metrics for the ranking and take the best solution.

The pseudocode of the scheduling algorithm is shown in Algorithm 1. The algorithm takes in input the conflicts $\text{Conflict}(e \rightarrow f)$, $e, f \in E$, the link load $\text{load}(e)$, $e \in E$, (measured as the number of packets to be transmitted by link e per frame) and the frame length T and returns the matrix schedule S . The row $S[e]$, with $e \in E$, is link e schedule: $S[e][t] = \text{TRUE}$ means that link e is scheduled to transmit at

time slot t (or, to put it in another way, node $e.\text{src}$ is scheduled to transmit to node $e.\text{dst}$ at time slot t).

Algorithm 1: Schedule Heuristic

```

input : Conflict relations  $\text{Conflict}(e \rightarrow f)$   $e, f \in E$ ; Link
         load  $\text{load}(e)$ ,  $e \in E$ , frame length  $T$ 
output : Node Schedule  $S$ 

1 begin
2    $E' \leftarrow \{e \in E | \text{load}(e) > 0\}$ ;
3   Bool  $M[[E]][T]$ ,  $S[[E]][T]$ ;
4   Real  $I_P[[V]][T] = N$ ; while  $E' \neq \emptyset$  do
5      $e_1 \leftarrow \text{Sort}(E', \text{metric})$ ;
6      $\text{sched}[\ ] \leftarrow \text{Available}(e_1, M, I_P, \text{load}(e_1))$ ;
7     forall  $i < \text{load}(e_1)$  do
8        $S[e_1][\text{sched}[i]] \leftarrow \text{TRUE}$ ;
9       forall  $f \in E'$  AND  $f \neq e_1$  AND  $\text{Conflict}(e_1 \rightarrow f)$ 
10        do
11           $M[f][\text{Mod}(\text{sched}[i] + c_{f, e_1}, T)] \leftarrow \text{TRUE}$ ;
12        end
13        forall  $v \in V$  AND  $v \neq e_1.\text{src}$  do
14           $I_P[v][\text{Mod}(\text{sched}[i] + d(e_1.\text{src}, v), T)] \leftarrow$ 
15             $I_P[v][\text{Mod}(\text{sched}[i] + d(e_1.\text{src}, v), T)] +$ 
16             $\text{Gain}(e_1.\text{src}, v)$ 
17        end
18       $E' \leftarrow E' - \{e_1\}$ ;
19    end
20  return  $S$ ;

```

As a preliminary step we first need to determine the link load $\text{load}(e)$, $\forall e \in E$. This is computed by considering the number of packets generated by each node $v \in V$ per frame and the routing. The algorithm then uses a simple greedy policy to schedule link transmissions. In the code, E' is the set of links yet to be scheduled which is initialized with the set of the links to be scheduled, i.e., the set of links with positive load (line 2). Then, until all link transmissions are scheduled, the algorithm iteratively sorts E' (the metrics for link ranking are discussed below) (line 5), picks the first link, e_1 , and schedules it (line 6) by determining the first $\text{load}(e_1)$ available slots. It then updates the schedule S (line 8) and the data structures which keep track of link conflicts (lines 9-11) and interference (lines 12-14).

The algorithm uses two matrices, M and I_P , to keep track of the link conflicts and interference due to all the other nodes transmissions at receiving nodes, respectively. For each link $e \in E$, $M[e]$ is a Boolean array which shows which are the unavailable time slots for transmissions. $M[e][t] = \text{TRUE}$ means that due to conflicts with *already* scheduled links, link e cannot transmit at time slot t . For each receiver $v \in V$, $I_P[v]$ is a real valued array which keeps track of the interference generated by the transmissions of the already scheduled links at receiving node v . Whenever a new link e_1 is scheduled, the two matrices are updated by adding which new links/slots are unavailable due to transmission over e_1 , and by adding the interference new transmission generates at other receivers. Observe that since propagation delays are non negligible we must account for them when determining which slots are affected by a given link transmission. For any link pair $e, f \in E$ for which a conflict exists, i.e., $\text{Conflict}(e \rightarrow f) = \text{TRUE}$,

we denote by $c_{e,f}$ the *conflict delay* between e and f , i.e., the delay between link e and link f transmissions that results into a conflict. In other words, if $c_{e,f}$ is the conflict delay between links e and f then a link e transmission at time $t+c_{e,f}$ conflicts with a link f transmission at time t . Conflict delays can be readily computed for the different types of conflicts described in Section III as follows.

Duplex conflicts. We have different type of duplex conflicts:

- 1) $e.dst = f.dst$ (Links e and f have the same destination):
A node cannot receive two different packets at the same time. The conflict delay is $c_{ef} = -c_{fe} = d_f - d_e$;
- 2) $e.src = f.src$ (Links e and f have the same transmitter):
A node cannot transmit two packets at the same time. In this case, $c_{ef} = c_{fe} = 0$;
- 3) $e.src = f.dst$ (Link e transmitter is link f receiver): A node cannot receive and transmit at the same time and $c_{ef} = -c_{fe} = d_f$.

Interference conflicts. In this case we consider the case when a link reception is interfered by another link transmission and the corresponding conflict delay is $c_{ef} = d_f - d_e$.

Algorithm 2: Function Available

```

input : link  $e$ ; matrices  $M$  and  $I_P$ ; number of packets to
        transmit  $n$ 
output : Node Schedule  $sched$ 

1 begin
2    $j \leftarrow 1$ ;
3   forall  $1 \leq i \leq n$  do
4     while  $\frac{P \cdot Gain(e.src, e.dst)}{I_P[e.dst, Mod(j+d_e, T)]} \leq SIR_{th}$  OR  $M[e][j]$  do
5        $j \leftarrow j + 1$ ;
6       if  $j > T$  then
7         it is not possible to schedule link  $e$  within  $T$ 
           slots;
8         return Error;
9       end
10      end
11       $sched[i] \leftarrow j$ ;
12       $j \leftarrow j + 1$ ;
13  end
14  return  $sched$ ;
15 end

```

The function Available (Algorithm 2) simply determines the first n available slots for a given link e . To this end, starting from the first slot in the frame, it simply scans the matrices I_P and M to determine whether transmission in slot j would either suffer interference from already scheduled links ($\frac{P \cdot Gain(e.src, e.dst)}{I_P[e.dst, Mod(j+d_e, T)]} \leq SIR_{th}$) or conflict with another link transmission ($M[e][j] = \text{TRUE}$) (lines 5-7); otherwise it schedules the link for transmission (line 11).

We observe that similarly to the optimal ILP model we need to choose the frame length T before executing the scheduling algorithm (we observe also that if the frame length is too short the heuristic might not be able to find a feasible solution). To this end, we follow the same approach taken for the optimal model and use binary search to identify the optimal frame length T^* .

The performance of the scheduling heuristic is heavily

affected by the ranking metric which ultimately determines the order according to which links are scheduled for transmission. The original work in [13] considered only a simple load metric $load(e)$. We investigated different alternative metrics which account for the additional following quantities:

- $conf(e)$, *number of links which conflict with link e transmission.* For a given link e we consider the number of conflicts generated by link e transmissions, i.e., the number of slots which would not be available to other links for transmission if link e is scheduled to transmit a packet. As a variation we can consider only the number of not yet scheduled links which would conflict with e . We will denote such number with $conf'(e)$;
- $free(e)$, *the number of still available slots for link e transmissions in the current frame.* Observe that this quantity must be recomputed every time a new link is scheduled.

We considered the following metrics:

- **LOAD** metric= $load(e)$. This is the metric proposed in [13]. Priority is given to links with more traffic to send. Ties are broken in favor of links with more conflicts;
- **LpFS** metric= $\frac{load(e)}{free(e)}$. Priority is given to the links with higher number of packets to transmit over free slot ratio. The idea is that priority should be given to links with less number of available free slots per packet;
- **LpFSC** metric= $\frac{load(e)}{free(e) \cdot conf(e)}$. Here priority is given to links with higher number of packets to transmit over the product of free slots and number of conflicts per transmitted packets. Here the idea is to favor links which have less free slots available and that cause less conflicts with other links per packet transmitted;
- **LpFSC'** metric= $\frac{load(e)}{free(e) \cdot conf'(e)}$. As above but conflicts are considered only with respect to links not already scheduled.

The idea is that the scheduling order should also account for the number of conflicts (and hence a reduced number of available slots for the other links) caused by a link transmission and/or the number of still available slots in the current frame. We also explored other combinations/metrics which exhibited worse overall performance. They have been thus omitted.

VI. PERFORMANCE EVALUATION

We have conducted a thorough set of experiments to evaluate the performance of the joint optimal routing/scheduling solution and of the heuristic. We implemented the ILP formulation of the optimization model in CPLEX solver, and the heuristics in MATLAB. Section VI-A describes the considered simulation scenarios and parameters. Results are presented in Section VI-B.

A. Simulation scenarios and parameters

The scenarios we have considered in our evaluation refer to oil well and pipeline monitoring and control applications. The sink is located on a surface buoy (mimicking the offshore platform), with the transducers at 10m depth. Two different network areas have been considered: 1 km \times 2 km and

1 km \times 4 km. They both have a rectangular shape, and force the packets to be routed through multiple hops. In both case the maximum node depth has been set to 250 meters. In the smaller area we have deployed 20 nodes resulting in an average route length of 1.7 hops, while in the larger area 40 nodes have been deployed requiring the nodes to make several 4 to 6 hops before having a packet relayed to the sink. The placement of the nodes is carried out by dividing the network area into a given number of sub-sections and by randomly assigning a node to each subsection, “wrapping around” the nodes that exceed the number of sub-sections. A grid of 3 \times 6 elements has been used when 20 nodes are considered and a grid of 3 \times 12 elements when 40 nodes are considered. As can be inferred from the above information, the number of nodes increases proportionally to the network area, hence the average density of the nodes is the same. In all scenarios the sink is assumed to be placed 500 m north and 500 m west of the bottom right corner of the network areas. To produce more accurate results, we consider the Bellhop Gaussian ray tracer [11] as transmission loss and propagation models when computing the signal attenuation over each link in the network. We connect to Bellhop via the WOSS [17] interface and use real historical environmental data for our computation. Environmental data such as sound speed profile and bathymetry profile have been obtained from real measurements. The environmental data we have considered refer to a location in the North of Europe, additional details are omitted as covered by a non disclosure agreement. Environmental noise is modeled using the empirical power spectral density equations reported in [16]. As parameters to those equations, we chose a moderate shipping factor of 0.5 and a wind speed of 7 m/s. The latter corresponds to level 4 of the Beaufort scale, which is typical of the considered region. The Bellhop ray tracing model, used with environmental data, provides a more accurate description of the underwater acoustic channel behavior, allowing us to compute the frequency-dependent acoustic path loss of each source-destination pair at a given location, as well as the spatially-varying interference induced by all active nodes. This information is provided as input to our analytical model and heuristic.

The parameters of the acoustic modem have been chosen to be in line with current commercial modems (see Table II).

We assume that each node can transmit a data packet at the beginning of each slot, the slot duration is equal to the time needed to transmit a packet. The summary of our simulation parameters and traffic rates are shown in Table II.

We have performed experiments considering different network traffic loads. We have considered the case where 25%, 50%, 75% and 100% of the nodes (with the exception of the sink) generate data packets. The nodes generating data packets are randomly selected and each of them generates (at most) one packet per frame.

B. Results

For every setup we perform experiments considering pairwise/binary and complete interference model. In Tables III, we

Table II
SIMULATION PARAMETERS

Bit rate	1000 bps
Carrier frequency	25.6 kHz
Bandwidth	4 kHz
Source Power Level (SPL) at 1 m	178 dB re μ Pa
Transmit / Receive / Idle power	3.3 W, 620 mW, 85 mW
Modulation	BPSK
Acquisition threshold	1 dB
Slot_duration	0.2 s
Sound speed	1500 m/s
Area sizes	1 km \times 2 km, 1 km \times 4 km
Number of nodes	20, 40
SNR_{th}	9 dB
SIR_{th}	6.5 dB

show results of evaluation considering topologies of 20 nodes, based on binary and complete interference model, while in Tables IV, we evaluate performance on 40 nodes topology. For the heuristics, we show the best results obtained with the different routing/ranking schemes we have considered. Given the low computationally complexity we found that in practice it is reasonable to run them all and take the best results.

The results show that the heuristic yields close to optimal results for both topologies and for all the different load levels. On average the heuristic estimation computes a minimum frame size which is 1.18 slots longer that the optimal value and a corresponding throughput reduction (node are assumed to generate one packet per slot) of 7%, with the heuristics running at a fraction of the cost, often within few seconds. Comparing the results for the different interference models, we observe that for the 20 node topology we obtain the same results for both the binary and the complete interference model, while for the 40 node topology we obtain longer frame durations, *i.e.*, lower throughput, when we consider the complete interference model. This suggests that in general using the simple binary interference model, as widely adopted in the literature, is just an approximation. Our experiments show that pairwise interference can be acceptable in small scenarios, where each single node transmission is able to interfere destructively with any other transmissions. On the other hand, in larger scenarios, with nodes further away from each other, and where the interference from a single far away node might not be sufficient to disrupt communication, the superposition of multiple distant nodes can still result in enough high interference to prevent packet reception. These phenomena cannot be captured using the simple binary interference model: As our results show using such approximation would provide optimistic interference evaluation and optimistic, but erroneous, results.

Table III
20 NODES TOPOLOGY - MINIMUM FRAME SIZE(SLOTS)

Binary interference model				
Traffic rate	25%	50%	75%	100%
Optimum	6	11	17	22
Heuristic	7	12	19	24

Complete interference model				
Traffic rate	25%	50%	75%	100%
Optimum	6	11	17	22
Heuristic	7	12	19	23

We now turn our attention on the performance of the

Table IV
40 NODES TOPOLOGY - MINIMUM FRAME SIZE(SLOTS)

Binary interference model				
Traffic rate	25%	50%	75%	100%
Optimum	12	21	33	44
Heuristic	13	23	34	45

Complete Interference model				
Traffic rate	25%	50%	75%	100%
Optimum	12	24	37	48
Heuristic	15	25	39	48

Table V
COMPLETE INTERFERENCE MODEL - MINIMUM FRAME SIZE(SLOTS)

20 nodes topology				
Traffic rate	25%	50%	75%	100%
<i>Load</i>	10	14	22	28
<i>LpFS</i>	10	13	21	23
<i>LpFSC</i>	7	13	23	23
<i>LpFSC'</i>	7	12	19	26

40 nodes topology				
Traffic rate	25%	50%	75%	100%
<i>Load</i>	22	31	48	59
<i>LpFS</i>	16	25	39	55
<i>LpFSC</i>	16	30	43	54
<i>LpFSC'</i>	16	25	41	48

heuristic under the different link ranking metrics. In Table V we summarize the results for the 20 and 40 node topologies, respectively. For the sake of simplicity we only report the results for the complete interference model. Comparing the different links ranking metrics we see there is no clear winner. However, *Load*, which is the basic ranking scheme proposed in [13] exhibits the worst performance, with overestimating the schedule length significantly on several problem instances. In general, the ranking metrics we proposed which combines load, number of free slots and conflicts generated yield good results, with *LpFSC'* and *LpFS* performing particularly well. Given the low computationally complexity of scheduling algorithm it is reasonable to just run them all and take the one with the best result. This has been done to derive the final schedule of the heuristic.

VII. CONCLUSION

Optimal traffic scheduling and routing are important features in UWSNs, as they can improve network throughput, energy consumption and latency. In this paper, we have presented an optimization framework, which exploits the periodicity nature of sensor traffic in UWSNs, and finds a periodic scheduling and routing solution that maximizes network throughput and minimizes energy consumption, while avoiding conflicts and interference between transmissions. For a more realistic interference model we improved the binary interference model, considering the combination of multiple transmissions that can disturb a packet reception. We also presented a heuristic solution which scales well with the problem size. Results show

that the gap between optimal model and heuristic prediction is minimal.

ACKNOWLEDGMENTS

This work has been supported in part by the European Commission under the 7th Framework Programme (grant agreement no. 258359 – CLAM).

REFERENCES

- [1] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: Research challenges," *Elsevier's Journal of Ad Hoc Networks*, vol. 3, no. 3, pp. 257–279, May 2005.
- [2] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "Challenges: Building scalable mobile underwater wireless sensor networks for aquatic applications," *IEEE Network, Special Issue on Wireless Sensor Networking*, vol. 20, no. 3, pp. 12–18, May 2006.
- [3] J. Heidemann, W. Ye, J. Willis, A. A. Syed, and Y. Li, "Research challenges and applications for underwater sensor networking," in *Proc. of the IEEE Wireless Communications and Networking Conference, WCNC 2006*, Las Vegas, NV, April 3–6 2006, pp. 229–235.
- [4] J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," in *Proceedings of the 1st ACM International Workshop on UnderWater Networks (WUWNet '06)*, Los Angeles, CA, USA, September 25 2006, pp. 17–24.
- [5] L. Lanbo, Z. Shengli, and C. Jun-Hong, "Prospects and problems of wireless communication for underwater sensor networks," *Wireless Communications and Mobile Computing, Special Issue on Underwater Sensor Networks*, vol. 8, no. 8, pp. 977–994, August 2008.
- [6] M. Kodialam and T. Nandagopal, "Characterizing achievable rates in multi-hop wireless multi-hop networks: The joint routing and scheduling problem," in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking, ACM MobiCom 2003*, San Diego, CA, September 14–19 2003, pp. 42–54.
- [7] V. S. A. Kumar, M. V. Marathe, S. Parthasarathy, and A. Srinivasan, "Algorithmic aspects of capacity in wireless networks," *SIGMETRICS Perform. Eval. Rev.*, vol. 33, no. 1, pp. 133–144, 2005.
- [8] A. Syed, W. Ye, and J. Heidemann, "Understanding spatio-temporal uncertainty in medium access with ALOHA protocols," in *Proc. of the third ACM International Workshop on UnderWater Networks (WUWNet '07)*, Montréal, Quebec, Canada, September 14 2007, pp. 41–48.
- [9] C. Petrioli, R. Petroccia, and M. Stojanovic, "A comparative performance evaluation of MAC protocols for underwater sensor networks," in *Proceedings of MTS/IEEE OCEANS 2008*, Quebec City, Quebec, Canada, September 15–18 2008.
- [10] S. Basagni, C. Petrioli, R. Petroccia, and M. Stojanovic, "Choosing the packet size in multi-hop underwater networks," *Technical Report 06/2009, Dipartimento di Informatica, Università di Roma "La Sapienza"*, Rome, Italy, June 2009.
- [11] M. et al., "Bellhop code." [Online]. Available: <http://oalib.hlsresearch.com/Rays/index.html>
- [12] R. Urick, *Principles of Underwater Sound*. McGraw-Hill, 1983.
- [13] C.-C. Hsu, K.-F. Lai, C.-F. Chou, and C.-J. K. Lin, "ST-MAC: Spatial-Temporal MAC scheduling for underwater sensor networks," in *Proceedings of IEEE INFOCOM 2009, the 29th IEEE International Conference on Computer Communications*, Rio de Janeiro, Brazil, April 19–25 2009.
- [14] Y. Guan, C.-C. Shen, and J. Yackoski, "MAC scheduling for high throughput underwater acoustic networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference, WCNC 2011, Quintana Roo, Cancun, Mexico, March 28–31 2011*, pp. 197–202.
- [15] L. Badia, M. Mastrogiovanni, C. Petrioli, S. Stefanakos, and M. Zorzi, "An optimization framework for joint sensor deployment, link scheduling and routing in underwater sensor networks," in *Proceedings of the First ACM International Workshop on UnderWater Networks, WuWNet 2006*, Los Angeles, CA, September 25 2006.
- [16] R. Coates, *Underwater Acoustic Systems*. New York: Wiley, 1989.
- [17] F. Guerra, P. Casari, and M. Zorzi, "World ocean simulation system (woss): a simulation tool for underwater networks with realistic propagation modeling," in *Proceedings of the Fourth ACM International Workshop on UnderWater Networks*, ser. WUWNet '09, Berkeley, California, USA, 3 November 2009, pp. 1–8.