

Optimized Packet Size Selection in Underwater Wireless Sensor Network Communications

Stefano Basagni, *Senior Member, IEEE*, Chiara Petrioli, *Senior Member, IEEE*, Roberto Petrocchia, *Member, IEEE*, and Milica Stojanovic, *Fellow, IEEE*

Abstract—In this paper, we investigate the effect of packet size selection on the performance of media access control (MAC) protocols for underwater wireless sensor networks, namely, carrier sense multiple access (CSMA) and the distance-aware collision avoidance protocol (DACAP). Our comparative analysis, conducted via ns-2 simulations, considers scenarios with varying, nonzero bit error rate (BER) and interference. We investigate metrics such as throughput efficiency (the ratio between the delivered bit rate and the offered bit rate), end-to-end packet latency, measured “per meter” to allow for different sizes of deployment areas, and the energy consumed to correctly deliver an information bit to the network collection point. Our results show the dependence of these metrics on the packet size, indicating the existence of an optimum. The optimum packet size is found to depend on the protocol characteristics, the bit rate, and the BER. For each protocol and scenario considered, we determine the packet size that optimizes throughput performance, and we show its effect on the normalized packet latency and on energy consumption.

Index Terms—Acoustic communications, media access control (MAC) protocols, packet size optimization, random access, underwater networks.

I. INTRODUCTION

INTEREST for undersea exploration and advances in underwater wireless modem technology motivate the investigation of underwater wireless sensor networks (UWSNs). Surveys such as those by Akyildiz *et al.* [1] and by Heidemann *et al.* [2] reveal that most of the existing solutions for UWSNs have addressed single-hop underwater topologies. More recently, the emphasis has shifted toward multihop networking as a means to deploy sensor nodes in a wider area as well as for increased efficiency [3], and research is active on different aspects of UWSNs.

Manuscript received August 17, 2011; revised January 17, 2012; accepted April 20, 2012. Date of publication June 12, 2012; date of current version July 10, 2012. This work was supported in part by the National Science Foundation (NSF) under Grants 0831728 and 0946610, the National Oceanic and Atmospheric Administration (NOAA) under Grant NA06OAR4170019 (2007-R/RT-2/RCM-21), the U.S. Office of Naval Research under Grant N00014-09-10700, and the European Union (EU) Project STREP FP7 CLAM (CoLIABorative eMbedded networks for submarine surveillance).

Associate Editor: J. Gomes.

S. Basagni and M. Stojanovic are with the Electrical and Computer Engineering Department, Northeastern University, Boston, MA 02115 USA (e-mail: basagni@ece.neu.edu; stojanovic@ece.neu.edu).

C. Petrioli and R. Petrocchia are with the Dipartimento di Informatica, Università di Roma “La Sapienza,” Rome 00198, Italy (e-mail: petrioli@di.uniroma1.it; petrocchia@di.uniroma1.it).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JOE.2012.2197271

The design of underwater media access control (MAC) and routing protocols for UWSNs has been considered in a number of publications [4]–[15]. However, only a few have been concerned with parameter optimization, and in particular with the choice of the packet size given a specific scenario and an application. The focus on packet size as a critical parameter of underwater communications stems from the investigation in [16], which addressed packet length optimization for maximizing throughput efficiency in a point-to-point (single-hop) scenario. That work shows that delay limitations of stop-and-wait MAC protocols in half-duplex acoustic channels can be avoided by careful selection of packet size. Packet size and its effect on the performance of underwater MAC protocols in multihop networks are investigated by Ng *et al.* [17]. The authors present an adaptation of the terrestrial multiple-access collision avoidance (MACA) protocol [18] to underwater acoustic channels. Despite the fact that this investigation considers only packets with preassigned sizes (150, 300, and 600 B), results show the remarkable impact that varying the packet size has on throughput. For the investigated packet sizes and the selected node deployment (a grid), it was shown that higher throughput is achieved with longer packets (within the range considered). This observation is consistent with the definition of MACA, which is based on a request-to-send/clear-to-send (RTS/CTS) handshake, and the fact that an ideal channel with zero bit error rate (BER) is assumed. An analytical framework for evaluating optimal packet size in multihop networks with forward error correction (FEC) is introduced by Vuran and Akyildiz [19]. The framework is applied to wireless terrestrial, underwater, and underground sensor networks by specializing signal attenuation to model a specific setting. To make the problem analytically tractable propagation delays are not taken into account, interference from nodes far away in the network is not considered, and results are shown only for the basic carrier sense multiple-access (CSMA) scheme.

In this paper, we investigate the impact of packet size on the performance of multihop communications in an underwater network. More specifically, given multihop scenarios with varying transmission rates and BERs, we determine the packet size that provides the best throughput efficiency (defined as the ratio between the delivered bit rate and the offered bit rate). We also analyze the impact of packet size on packet latency (measured “per meter” to unify the performance over varying deployment areas) and on energy consumed for each bit correctly delivered to the network data collection point (the *sink*). We consider two underwater MAC protocols, namely, CSMA and the distance-aware collision avoidance protocol (DACAP) [8], which exemplify

MAC schemes with and without the RTS/CTS handshake for collision avoidance, respectively. Our choice is motivated by a comparative performance evaluation among several MAC protocols that we performed in [9], which showed that these two protocols are the best performing in the multihop scenario.

In a multihop network, the noise and fading, which result in nonzero BER, are not the only cause of packet loss. Here, interference is another important factor that contributes to performance degradation. The situation is exacerbated in acoustic scenarios where the low spreading factor (path loss exponent) supports interference from nonneighboring nodes, and even from those that are far away in the network. Through extensive simulations on most of the underwater MAC protocols proposed so far, we have observed that the vast majority of collision-induced packet losses are due to this latter type of interference [9]. Specifically, in the case of RTS/CTS-based access, we observed that 90% of packet losses are due to interference coming from nodes that are outside the receiver's transmission range. This effect occurs even in networks where the traffic is not particularly high. Moreover, many of these collisions occur between control and data packets. The latter kind of collisions could be completely avoided by adopting out-of-band signaling, i.e., by using different channels for control and data packets. However, as we showed in [20], splitting the available bandwidth affects performance negatively, increasing source-to-sink packet latency and being barely effective in conditions of high BER (which may be typical of an underwater system).

The multihop scenarios we investigate are challenging in that we consider a relatively large number (up to 100) of nodes randomly deployed over an arbitrary shallow-water area, and data generation rates corresponding to different application requirements. We expect this to be the core scenario of future underwater network deployments, where further components could be added, such as mobile unmanned devices, or support for underwater cellular-like architectures. We also investigate different network sizes (16, 35, and 100 nodes), topologies (single-hop and multihop), and deployment areas, and discuss how these parameters affect packet size selection. Results are obtained through ns-2-based simulations [21] combined with the Bellhop ray tracer [22] for modeling the acoustic channel propagation. The Bellhop ray tracing model is used with real environmental data that provide us with a first approximation of the underwater acoustic channel behavior. As preliminarily shown by Stojanovic [16] for throughput in single-hop communications, our work confirms that crucial metrics such as throughput efficiency, latency, and energy consumption in multihop UWSNs can be greatly enhanced by a judicious choice of the packet size. It is also confirmed that the best packet size depends on the data generation rate, the bit rate, and the BER. Our results provide practical insights on designing MAC protocols for multihop UWSNs and for choosing the best packet size given specific scenarios and application requirements. They also show that there are packet sizes that should not be used with channel access methods similar to those investigated here, in the sense that they result in poor network performance regardless of the transmission rate and the BER. These considerations are particularly important for practical system deployments with existing acoustic modem technology.

The remainder of the paper is organized as follows. In Section II, we briefly describe the protocols investigated in this paper, namely, CSMA and DACAP. Section III presents the scenarios that we use to assess the network performance. Performance results are described in Section IV for the main scenario (networks with 100 nodes) and in Section V, where we vary the network size, topology, and the deployment area. Section VI concludes the paper.

II. CSMA AND DACAP

In CSMA [18], when a node has a data packet to transmit, it first checks whether the channel is idle or busy. If the channel is idle, it starts the packet transmission. If the channel is busy, the node delays the transmission according to the CSMA exponential backoff mechanism. Acknowledgment (ACK) packets can be used to add robustness. If the ACK is not received within a given time (set to $2\text{Delay} + \text{ackTime}$, i.e., twice the propagation delay plus the time needed to transmit the ACK), the data packet is retransmitted, either until successful reception, choosing the backoff time of each retransmission in an interval twice as long as the previous one, or until the maximum limit of retries (maxRetries) has been reached. The value of Delay is initially set to an upper bound of the maximum propagation delay, maxDelay (computed based on the node maximum transmission range), and successively set to half the time difference between the packet transmission time and the time of the reception of its ACK. The backoff time is chosen uniformly at random in $[0, T]$, where $T = 2^{\text{txRetry}}(2\text{maxDelay} + \text{dataTime} + \text{ackTime})$, and dataTime is the time needed to transmit a data packet. In the CSMA version without ACKs ackTime is assumed to be zero. If an idle node overhears a data packet in the channel (and ACKs are used) it backs off, thus allowing the transmitter to correctly receive the ACK and enabling the receiver to forward the data that it has just received. A node that just received an ACK backs off to allow the destination to forward the data, and to let the other nodes (if any are trying) to access the channel.

DACAP [8] uses the RTS/CTS handshake for reserving the channel for packet transmission, enriching this common mechanism with a method to accommodate the longer delays of underwater links. A node that has a data packet to send checks the status of the channel. If the channel is idle, it transmits an RTS. If the channel is busy, the sender computes a backoff time and after this time checks the channel again. Upon correctly receiving an RTS packet, a destination node replies immediately with a CTS. It then waits for the data packet, which can be acknowledged or not, depending on the chosen version of the protocol [8]. If while waiting for a data packet a destination node overhears a control packet intended for some other node, it sends a very short WARNING packet to its sender, to alert it about possible interference that could affect the upcoming communication. Upon receiving a CTS packet, the sender waits for a time T_{warning} before transmitting the data packet. The time T_{warning} is defined as the minimum time allowing neighboring nodes not to interfere. Its computation is dependent upon the propagation time between the source and the destination (estimated by the sender through the RTS/CTS handshake) and on other factors concerning the distance of potential interferers. If

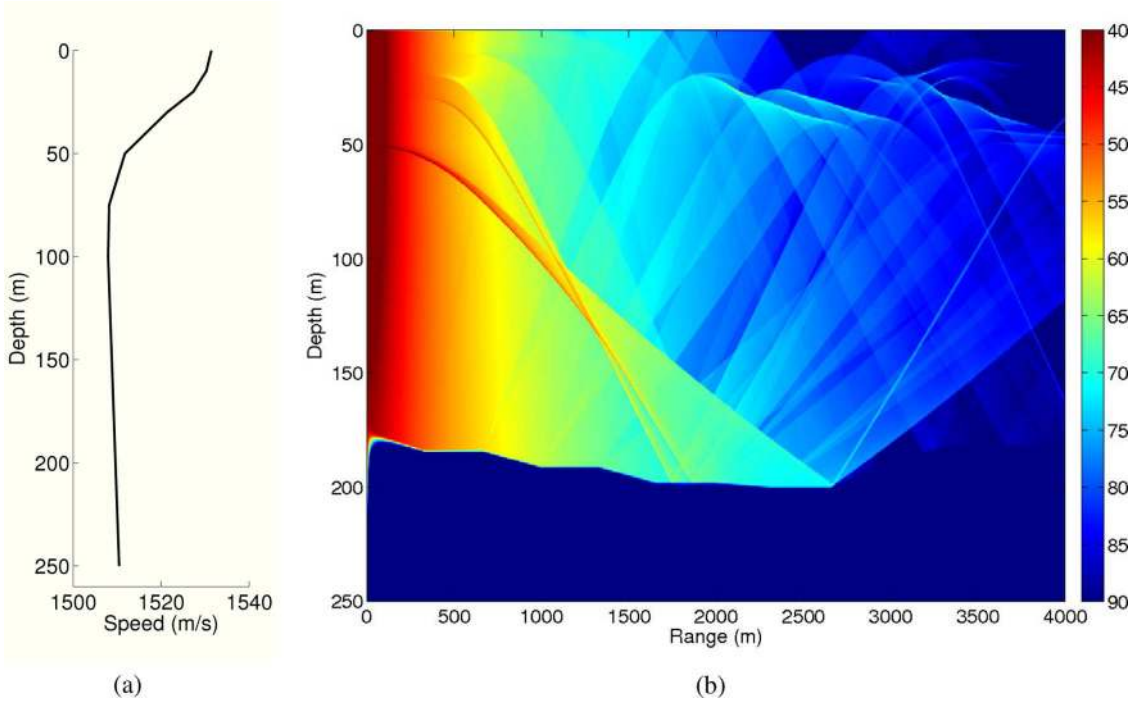


Fig. 1. Characteristics of the deployment area: (a) SSP and (b) acoustic field (incoherent) measured in dB re 1 μ Pa at 1 m.

while waiting for a CTS the sender overhears a control packet, it aborts the transmission. It also aborts the transmission if it receives a WARNING packet from the destination while during T_{warning} , or if it overhears a control packet from another node. In these cases, the sender computes a backoff time and tries again later (for a predefined number of times). Since a receiver that sent a WARNING packet does not know if this packet had reached the sender in time to make it abort the transmission, it continues to listen to the channel because the data packet may still be received correctly.

In the case of DACAP with ACKs, the sender backs off and retries if no ACK is received after data transmission within a specified time. The same happens if while waiting for the ACK the sender instead overhears an RTS, a CTS, or a DATA packet from other nodes.

In the present analysis, we implemented CSMA and DACAP with ACKs. We found these choices to be the best in terms of packet delivery ratio and latency performance [9].

III. SYSTEM MODEL

We have implemented CSMA and DACAP using ns2-MIRACLE [23] on top of ns-2 [21], connected to the Bellhop propagation simulator [22] via the World Ocean Simulation System (WOSS) interface [24]. Bellhop allows 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. The ray tracing model is used with real environmental data corresponding to a location in the Mediterranean sea off the coast of the Pianosa island (Tuscan archipelago), with the coordinate (0, 0, 0) of the surface located at 42°32'0" N and 10°22'0" E. In particular, we used the sound-speed profiles (SSPs), bathymetry, and information on the type of bottom sediments of the selected area, obtained from the World Ocean Database [25], from the

General Bathymetric Chart of the Oceans (GEBCO) [26], and from the National Geophysical Data Center's Deck41 database [27], respectively. Fig. 1(a) shows the SSP and Fig. 1(b) shows the related acoustic field. The SSP is retrieved by WOSS from the World Ocean Database (average of measurements from September 2009). The acoustic field is obtained through Bellhop ray tracing for a signal source located at a depth of 50 m. The bottom type is clay and silt.

A. Simulation Scenarios and Settings

Parameter setting as well as the characteristics of selected topologies are shown in Table I. We consider networks with 100 nodes (99 nodes plus the sink) placed in a region with 4-km \times 4-km footprint. Nodes are placed uniformly at random at different depths, ranging from 20 to 100 m. Every node has an average of 15 neighbors. The sink is placed centrally on the surface with the transducer 10 m below.

Packets are transmitted from the nodes to the sink through predetermined shortest routes. Each packet that makes it to the sink traverses an average of 2.5 hops (the maximum number of hops is 4).

We considered three bandwidths, namely, 200, 2000, and 20 000 Hz. Bandwidth efficiency is set to 1 b/s/Hz and we assume binary phase-shift keying (BPSK) modulation. The carrier frequency is 24 kHz for bandwidths of 200 and 2000 Hz, and 22 kHz for the third bandwidth. For each value of the bandwidth, we have computed the transmission power P_{tx} that results in BERs on each route equal to 10^{-4} or 10^{-6} . Specifically, $P_{tx} = 139, 149, \text{ and } 159$ dB for BER = 10^{-4} and $B = 200$ Hz, $B = 2000$ Hz, and $B = 20\,000$ Hz, respectively; similarly, $P_{tx} = 142, 152, \text{ and } 162$ dB for BER = 10^{-6} . All the considered transmission power values are expressed in dB re 1 μ Pa at 1 m. Traffic is generated according to a Poisson process with aggregate (network-wide) rate of λ packets per

TABLE I
SIMULATION PARAMETERS AND TOPOLOGY PROPERTIES

Deployment area	4km × 4km
Water depth	100m
Latitude of coordinate (0,0,0)	42° 32' 0" N
Longitude of coordinate (0,0,0)	10° 22' 0" E
Bandwidth (B)	200Hz, 2000Hz and 20000Hz
Bandwidth efficiency	1 b/s /Hz
Bit rate (R_b)	200 b/s, 2000 b/s and 20000 b/s
Packets per packet time ($\bar{\lambda}$)	0.01, 0.1, 0.2, 0.3 and 0.6
Data payload (N_b)	100B and from 200B to 3000B in increments of 200B
Synchronization peering at the physical layer	10ms
Buffer size at each node	50 packets
Detection threshold	1dB
Reception and idle power	0.1 W
Transmission power (P_{tx}) for BER = 10^{-4}	$P_{tx} = 139\text{dB}$, $P_{tx} = 149\text{dB}$ and $P_{tx} = 159\text{dB}$
Transmission power (P_{tx}) for BER = 10^{-6}	$P_{tx} = 142\text{dB}$, $P_{tx} = 152\text{dB}$ and $P_{tx} = 162\text{dB}$
Carrier frequency	24kHz ($B = 200, 2000\text{Hz}$); 22kHz ($B = 20\text{kHz}$)
Number of nodes (N)	100
Average number of neighbors reached	15
Average (maximum) number of hops per route	2.5 (4)

second. Once a packet is generated, it is associated with a source selected randomly among all the nodes. The destination of all packets is the sink. We define the normalized packet rate as $\bar{\lambda} = \lambda T_{\text{pack}}$, whose values are considered in the range from 0 to 1 packets per packet time. The packet duration is $T_{\text{pack}} = N_b/R_b$, where N_b is the packet size in bits and R_b is the bit rate. Simulation results presented here concern very low traffic ($\bar{\lambda} = 0.01$), low traffic ($\bar{\lambda} = 0.1$), medium traffic ($\bar{\lambda} = 0.2, 0.3$), and high traffic ($\bar{\lambda} = 0.6$). (We will refer to the packet length in bits as packet size, and to the packet length in seconds as packet duration.) Results from simulations with very low traffic are shown only for scenarios where the nodes transmit at 2000 and 20 000 b/s, while results for high traffic are shown only for scenarios where the nodes transmit at lower bit rates. This pairing is made because traffic is normalized, and, as a consequence, the actual number of packets injected into the network for a given $\bar{\lambda}$ increases with the bit rate. When $\bar{\lambda} = 0.6$ and the bit rate is 2000 b/s or higher, the network becomes congested, and performance is considerably degraded.

To assess the impact of packet size on the protocol performance, we consider data packet payloads of 100, 200, 400, 600, ..., 2800, 3000 B (for a total of 16 different packet sizes). The total size of a data packet is given by the payload plus the headers added by the different layers (physical through network). The physical-layer header contains all the information needed by the modem to correctly start receiving a packet (synchronization preamble, delimiters, etc.). At the physical layer, nodes need a synchronization peering time which is taken to be on the order of 10 ms (the physical header overhead changes according to the data rate). The MAC header contains the sender and the destination IDs, and the packet type. Its length is set to 3 B. The sizes of RTS and CTS packets are set to 6 B, and ACK and WARNING packets are 3 B. To correctly detect each packet (control or data) the detection threshold at the receiver is set to 1 dB, which is the threshold used by the Woods Hole Oceano-

graphic Institution (WHOI, Woods Hole, MA) micromodems [28]. Packets received in error because of channel distortions (modeled by a nonzero BER), collisions, and interference are discarded. We do not consider packet loss due to malfunctioning hardware or inaccurate synchronization. Each node limits the number of packets that can be stored to 50. Whenever the buffer is full and a new packet arrives, the oldest packet is discarded. Our implementation of CSMA mandates discarding of a packet after seven attempts of either accessing the channel or retransmitting the packet. The same holds for DACAP concerning RTS packets. For data packets, only four attempts of either accessing the channel or retransmitting are made before the packet is discarded (values tuned through simulations). For both protocols, we consider the version with ACKs, which proved more robust, especially in multihop scenarios.

B. Simulation Metrics

Effectiveness and costs of delivering bits to the sink are assessed through the following metrics.

- *Throughput efficiency*, defined as the ratio between the average bit rate successfully delivered to the sink and the average offered bit rate $N_b \bar{\lambda}$.
- *End-to-end latency per meter*, defined as the time between the packet generation and the time of its correct delivery to the sink, divided by the distance between the source and the destination. Normalization by distance is used so as to unify the performance over varying deployment areas (a larger area will entail proportionately larger propagation delays). This metric is computed only for the packets correctly delivered, and averaged over all such packets. A protocol that keeps this metric constant for a varying deployment area can be considered scalable.
- *Energy per bit*, defined as the energy consumed by the network to correctly deliver a bit of data to the sink.

IV. PERFORMANCE RESULTS

Results on the three metrics defined above are shown in Figs. 2–10. Every point reported in the figures has been obtained by averaging over the number of simulations runs needed to achieve a statistical confidence of 95% with a 5% precision.

A. Throughput Efficiency

Fig. 2 shows the results for the bit rates considered when the BER is 10^{-6} . For both CSMA and DACAP, the throughput efficiency steadily increases with the packet size, reaching a maximum that depends on the offered load. The lower throughput efficiency for shorter packets is due to the overhead imposed by control packets. The overhead particularly affects DACAP, which uses RTS and CTS packets in addition to the ACKs.

In all the cases, as traffic load increases the throughput efficiency decreases. This effect is due to multiple reasons. When the number of packets is higher, the nodes are more likely to find the channel busy. Moreover, and more significantly, the chances of collisions are higher, and the corresponding retransmissions degrade the throughput. This is especially true in a multihop scenario where each hop generates extra data packets; new overhead (control packets) and collisions also occur because of interference generated by nodes that are multiple hops away. Overall, the number of transmission attempts doubles (triples) for CSMA (DACAP) when $\bar{\lambda}$ increases from 0.01 to 0.2 at $R_b = 20\,000$ b/s. This increase is more contained (only around 50%) for the lowest data bit rate considered.

CSMA outperforms DACAP in all scenarios (blue lines versus red ones, respectively, in color; dotted lines versus steady lines in black and white). This advantage occurs because of CSMA lower access time, i.e., the lack of control handshake used by DACAP. The use of RTS and CTS affects DACAP especially for short data packets, and when the propagation delay is overwhelming with respect to the transmission delay, which makes the handshake duration particularly long.

Considering the same normalized packet rate $\bar{\lambda}$, the higher the data rate, the higher the amount of bits correctly delivered to the sink, but the lower the throughput efficiency. This effect occurs because increasing the bit rate implies increasing the number of data packets injected into the network. For instance, increasing the bit rate from 200 to 2000 b/s increases the number of packets ten times. Each of these packets takes one tenth of the transmission time that it took for transmitting it at 200 b/s, so that the total transmission time stays the same. However, since the propagation delay (which remains the same) is now present for each of the extra packets, the time needed to correctly deliver each packet is longer, and the channel utilization correspondingly lower.

As the BER increases to 10^{-4} , the situation changes considerably, as shown in Fig. 3. The throughput no longer increases steadily with the packet size. It reaches a maximum and decreases thereafter. Given the high BER, longer packets suffer from a higher probability of being corrupted during transmission and therefore require retransmission. The value of the maximum throughput depends on the offered load and on the bit rate. For example, when $\bar{\lambda} = 0.1$ and $R_b = 2000$ b/s, the maximum achievable throughput is about 65% for CSMA and 45% for

DACAP, which is quite a decrease from the 97% seen at BER of 10^{-6} . The desired range of operation is in the stable region (to the left of maximum), i.e., with packets slightly shorter than the optimum.

The above results clearly show the sensitivity of throughput to the packet size. Looking at scenarios with $\bar{\lambda} = 0.3$ and $R_b = 200$ b/s, where the optimal packet size is about 200 B for CSMA and 400 B for DACAP, we note that choosing 1400-B-long packets would result in a throughput efficiency of only 25%. This is a significant loss compared to the optimal 50% and 40%, which emphasizes the importance of careful packet size selection. A comparison of the results depicted in Fig. 3(a)–(c) confirms that the overall performance is also affected by the bit rate, as in the case of low BER.

Based on these results, rough guidelines can be suggested for the design of practical systems, by showing which packet size optimizes throughput efficiency in which scenario. Results are shown in Fig. 4. Ties, if any, are broken by packet latency per meter and energy consumption values. It is clear that DACAP, being more affected by the propagation delay, shows the best performance with larger packet sizes. Not having to endure extra delays for accessing the channel, CSMA instead prefers short data packets when the traffic load is low. As the traffic load increases, larger packets result in a lower number of channel accesses. This fact explains why the maximum throughput efficiency is achieved for larger packets.

B. Latency

Fig. 5(a)–(c) shows the average packet latency per meter in networks with varying bit rates and BER of 10^{-6} . As expected, the lower is the traffic, the lower is the latency. In almost all the cases (with the exception of very low traffic for both protocols and low traffic for CSMA) the two protocols incur high latency per meter when the packets are small. As their size increases, the normalized latency decreases, reaches a minimum, and then starts to increase again. The reason for such behavior is twofold. 1) When shorter packets are used, more packets are injected into the network for the same $\bar{\lambda}$, resulting in more collisions and therefore more retransmissions. Each retransmission incurs a high propagation delay, resulting in higher latency per meter. 2) Longer packets result in longer transmission delays, which causes longer latency per meter. The contribution of transmission delay to the latency is particularly relevant at low bit rates, as shown by the significant increase of the latency per meter with the packet size [Fig. 5(a)]. In general, latency per meter is higher for DACAP than for CSMA whenever the propagation delay is significantly longer than the transmission delay. In this case, DACAP pays a price for the RTS/CTS handshake. When the ratio between the transmission and propagation delay is much greater than 1, the reservation approach used by DACAP pays off, as evident in its lower latency per meter for most packet sizes in Fig. 5(a). Decreasing the ratio between the two delays decreases the latency per meter, as shown in Fig. 5(b) and (c) (“zoomed” on those values that correspond to acceptable throughput). Latency per meter at $R_b = 20\,000$ b/s is higher than when $R_b = 2000$ b/s because of the higher number of packets injected into the network, which build up congestion with an immediate impact on latency.

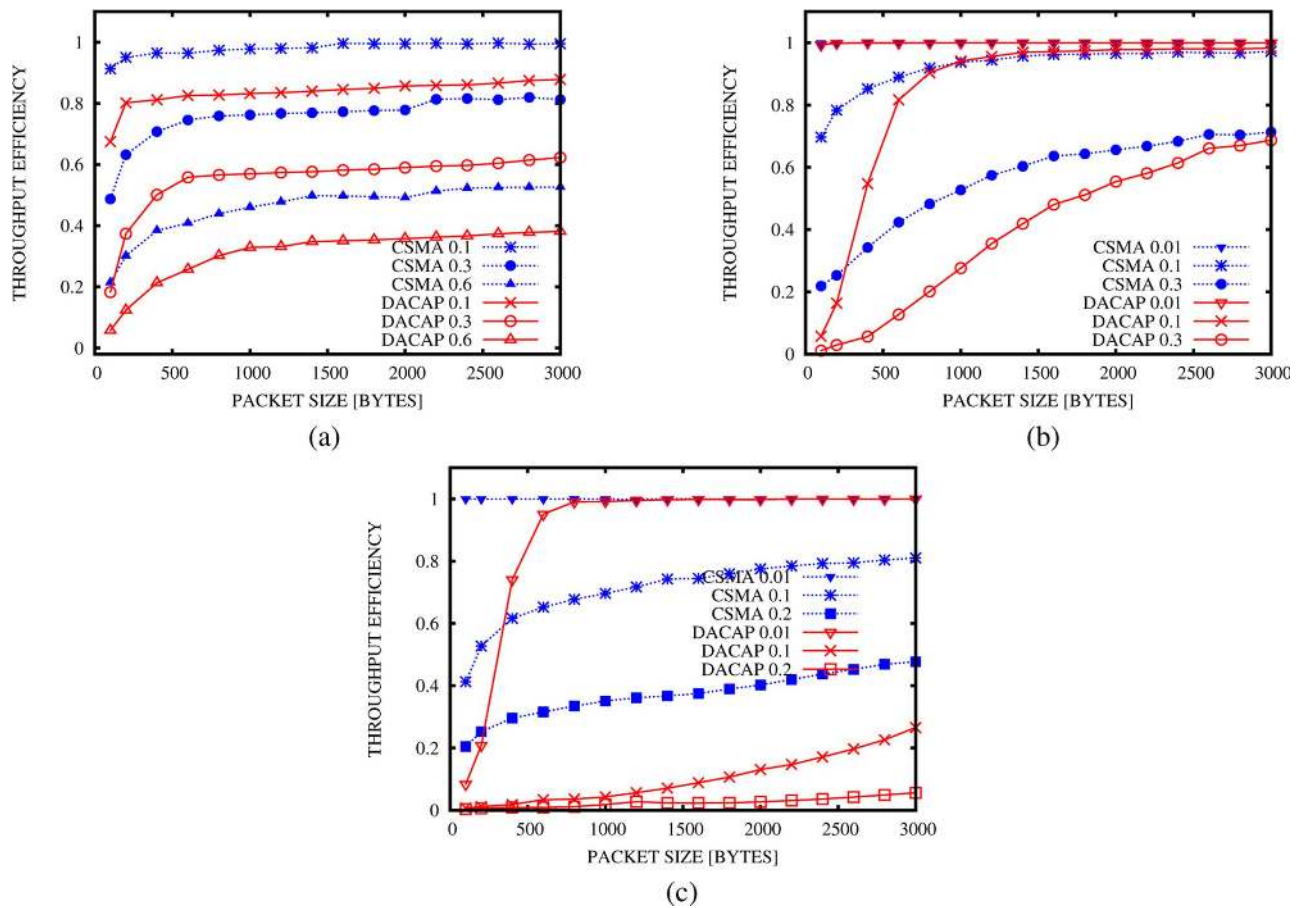


Fig. 2. Throughput efficiency for different bit rates R_b and $\text{BER} = 10^{-6}$. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20\,000$ b/s.

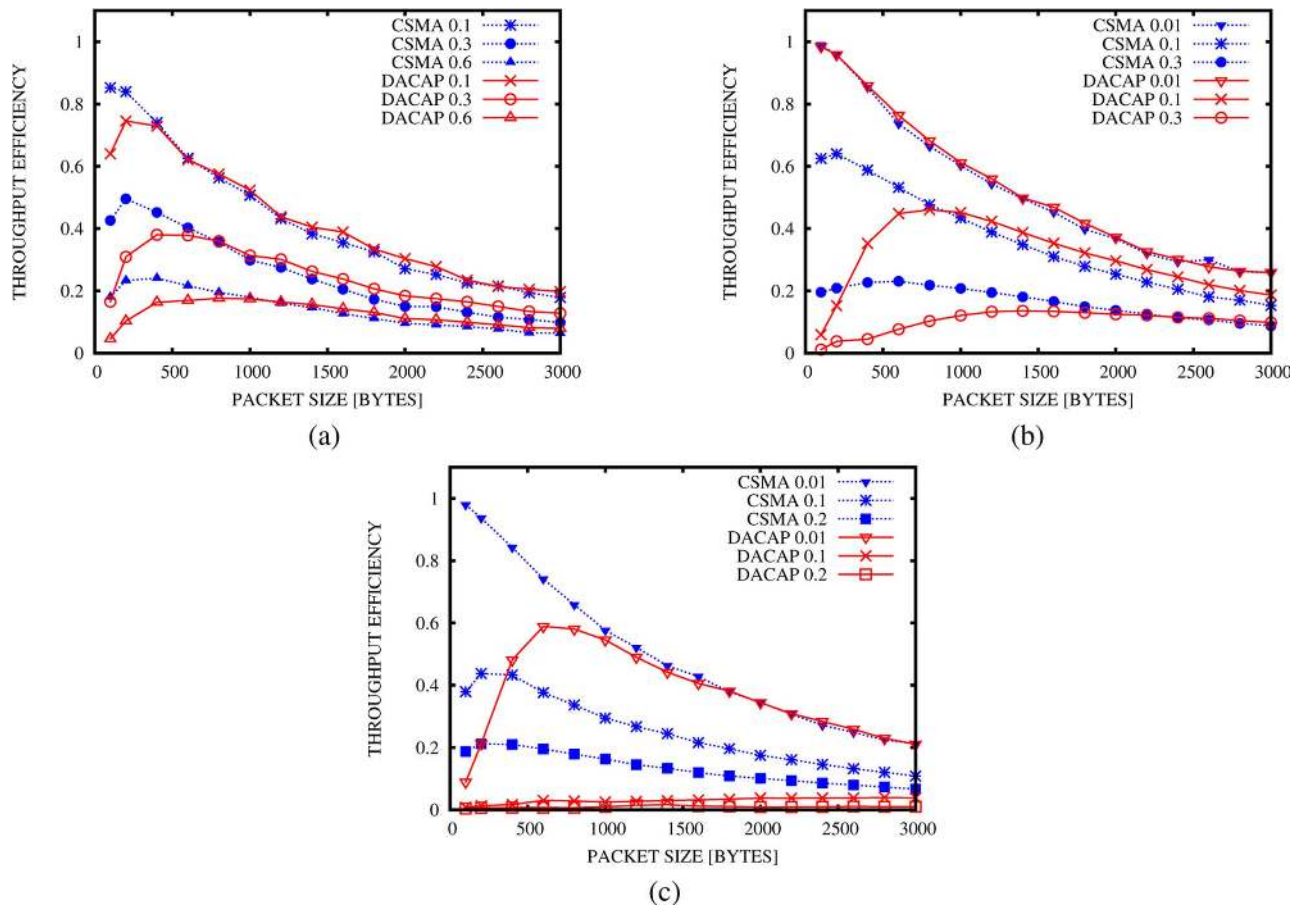


Fig. 3. Throughput efficiency for different bit rates R_b and $\text{BER} = 10^{-4}$. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20\,000$ b/s.

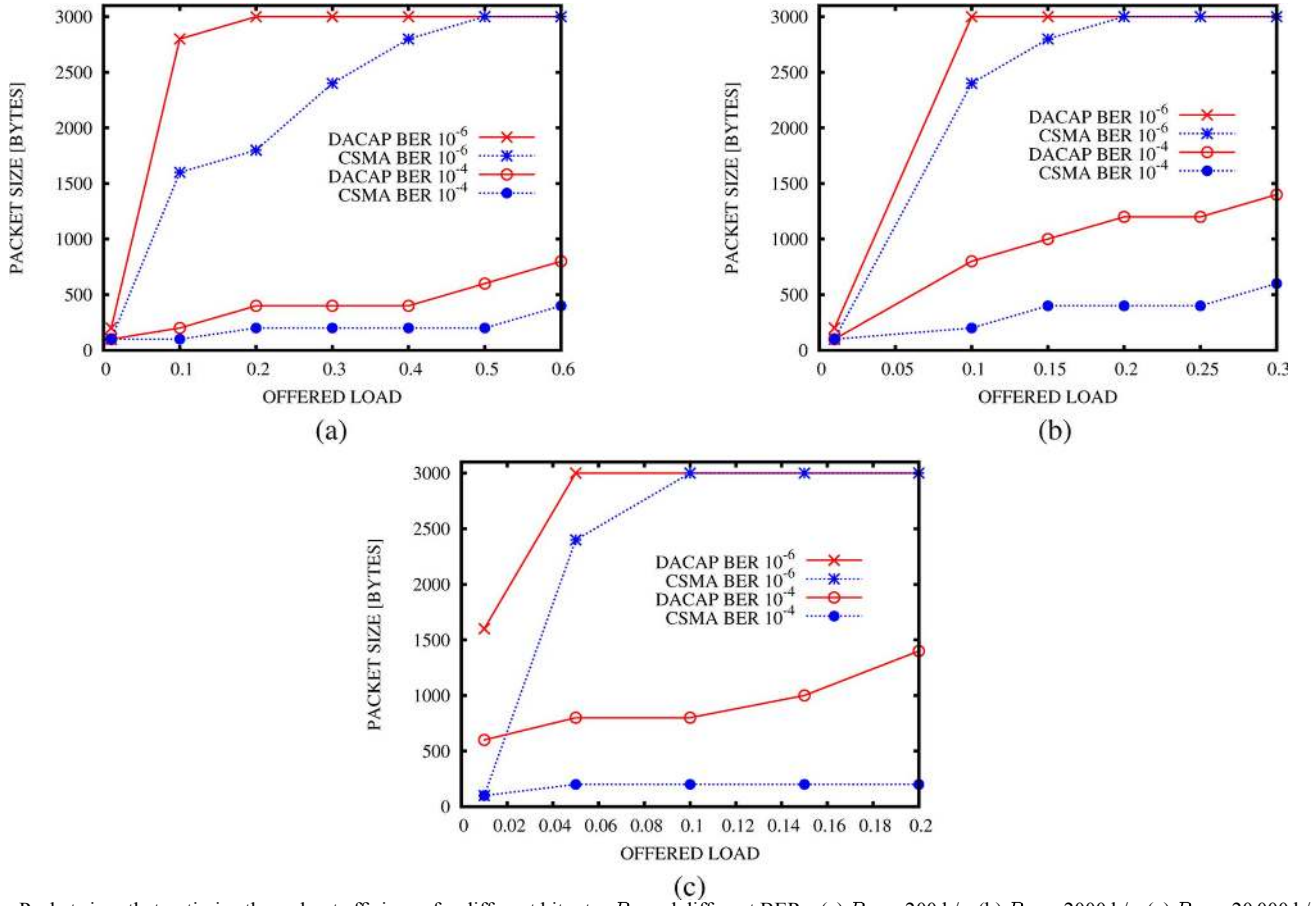


Fig. 4. Packet sizes that optimize throughput efficiency for different bit rates R_b and different BERs. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20\,000$ b/s.

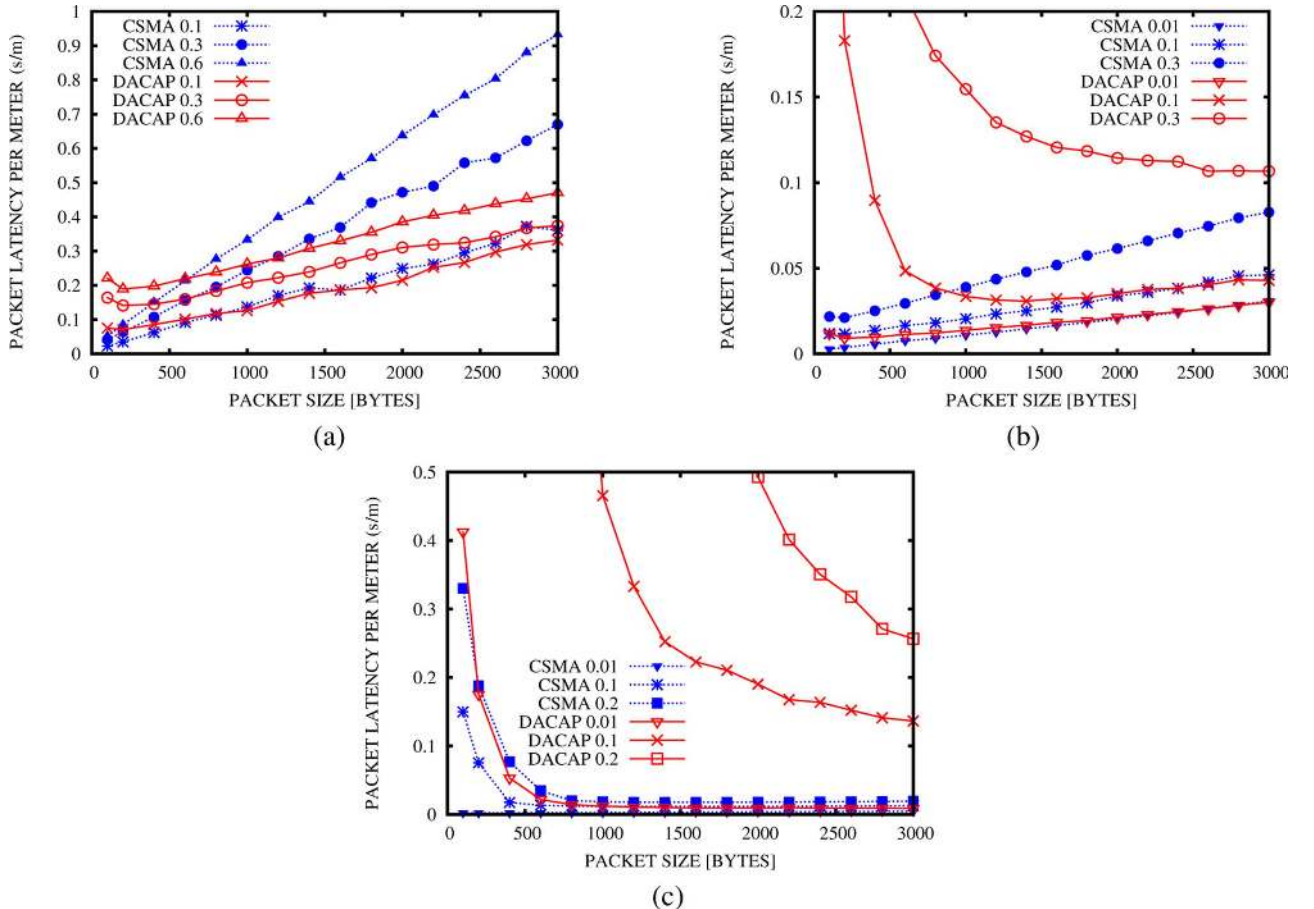


Fig. 5. Latency per meter for different bit rates R_b and $\text{BER} = 10^{-6}$. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s, zoom. (c) $R_b = 20\,000$ b/s, zoom.

To provide further insights on latency, and to show more clearly its sensitivity to the ratio between transmission and propagation delay, we investigate the different components of the packet latency per meter for both CSMA (Fig. 6) and DACAP (Fig. 7). Each figure shows the delay composition for increasing traffic load and bit rate. Results shown here concern three different packet sizes: short (100 B), medium (1400 B), and long (3000 B).

Packet latency components for CSMA are the packet propagation delay (Propagation), the time each data packet stays in the queue before transmission (Queue), the transmission delay (DataTx), and the time spent in backoff for missed ACK reception or for finding the channel busy when trying to access it (BackoffData). Once the packet size has been fixed, increasing bit rates correspond to shorter transmission delays (the propagation delay remains the same). Clearly, the contribution of packet transmission to latency is reduced, while that of propagation delay is increased. The case with $R_b = 2000$ b/s, very low traffic ($\bar{\lambda} = 0.01$), and medium to large packet sizes [first bar of the middle triplet in Fig. 6(b) and (c)] may seem to contradict the trend we just explained. In fact, since the traffic is very low, there are no obstacles in accessing the channel, and the latency is only due to the transmission and propagation delays. Since the size of the packet matters, the transmission component (in percent) is dominant. The contribution of the BackoffData component is negligible. In all other cases, instead, we observe that accessing the channel is always challenging, as demonstrated by the large BackoffData contribution to the latency. As expected, the queuing delay component of latency increases with the data rate R_b , because of the higher number of packets in the network. While this trend is maintained for all packet sizes, we observe that when packets are small, their time in the queue is particularly high. This observation offers evidence that congestion builds up, as confirmed by the results on throughput efficiency shown in Fig. 2. Specifically, the cases depicted in Fig. 6(a), where the queuing delay is overwhelming with respect to all other latency components ($N_b = 100$, $R_b = 20\,000$ b/s, and $\bar{\lambda} = 0.1$ and 0.2), correspond to CSMA throughput efficiency that is always below 40%.

Fig. 7(a)–(c) illustrates the results for DACAP. Latency components for DACAP are the same as for CSMA, plus the time for RTS/CTS transmission (RtsCtsTx), the time spent in backoff for RTS packets (for missed CTS reception) or for finding the channel busy while trying to access it (BackoffRts), and the warning time (T_{warning}). We observe that when using control packets to reserve the channel, the backoff delay due to missed ACKs or to the channel being busy is almost negligible. In these cases a high percentage of delay is for RTS/CTS propagation and backoff. While the relative impact of transmission and propagation delays on latency per meter has trends similar to those of CSMA, we observe that in many scenarios DACAP queuing delay has a noticeably more prominent role. In these scenarios, the toll imposed by the RTS/CTS exchange is high because of the propagation delay, and the reward of limiting collisions to the shorter RTS and CTS packets is not enough to compensate for it.

Packet latency per meter for a network with $\text{BER} = 10^{-4}$ is shown in Fig. 8. We notice a performance similar to the case

of lower BER. The observations made for that scenario largely hold for this one as well. Latency per meter is slightly higher because of the large number of retransmissions, especially for longer packets. Although not emphasized in the figures because of the normalized metric, we note that the packets that contribute to the latency per meter are mostly generated by nodes closer to the sink as packets from farther nodes are discarded because of reaching the maximum number of retransmissions. Those from closer nodes make it to the sink. However, because of the high BER, their successful delivery occurs after many retransmissions, which causes an increase in latency.

C. Energy

The final set of simulations concerns the energy spent to deliver a bit of data correctly to the sink. We start by showing the results concerning the case where nodes are always active, i.e., by considering the energy spent for transmitting and receiving a bit, as well as that spent when a node just listens to the channel (idling). Results are shown in Figs. 9 and 10 for scenarios with $\text{BER} = 10^{-6}$ and $\text{BER} = 10^{-4}$, respectively.

As is typical of wireless communications, the greater part of the energy is spent on just listening to the channel. In all considered scenarios, we observed that the time each node spends idle is on average two orders of magnitude higher than that spent on transmitting. For instance, in scenarios with $\text{BER} = 10^{-6}$ and low traffic ($\bar{\lambda} = 0.1$), regardless of R_b , a node spends more than 97% idling, leaving the remaining 3% of the time for transmission and correct reception. Increasing the traffic changes these values only slightly: At the highest load considered for each bit rate, with either CSMA and DACAP, the nodes stay idle for more than 95% of the time. This is because the results are obtained by averaging among all the nodes in the network, and even if the nodes close to the sink might be congested, the majority of the nodes at the fringe of the network have little to do.

Given that idle listening is the dominant cause of energy expenditure, the total energy consumption in different scenarios does not change considerably. However, energy per bit varies remarkably, reflecting the fact that depending on the scenario, the network is able to deliver very different amounts of traffic to the sink correctly. For a given $\bar{\lambda}$, the higher the bit rate, the more bits are delivered, and the lower the energy per bit is, as can be seen by comparing the results of Fig. 9(a) to those of Fig. 9(c). This fact also explains the better energy performance for those ranges of packet sizes for which throughput efficiency is the highest. The performance of DACAP degrades at higher bit rate [Fig. 9(c)], which can be attributed to its lower throughput efficiency in that scenario.

Results for $\text{BER} = 10^{-4}$ are shown in Fig. 10. The same considerations made for the lower BER apply here as well. For both protocols and for all the bit rates, because of the lower number of bits correctly delivered due to the higher probability of error, the energy spent to deliver those bits increases with the packet size. The actual trends shown in the figures are different from those of Fig. 9 and correspond to the different trends observed for the throughput efficiency.

To overcome the time and the energy expenditure in idle state, acoustic nodes can be endowed with a “wake-up” capability, by

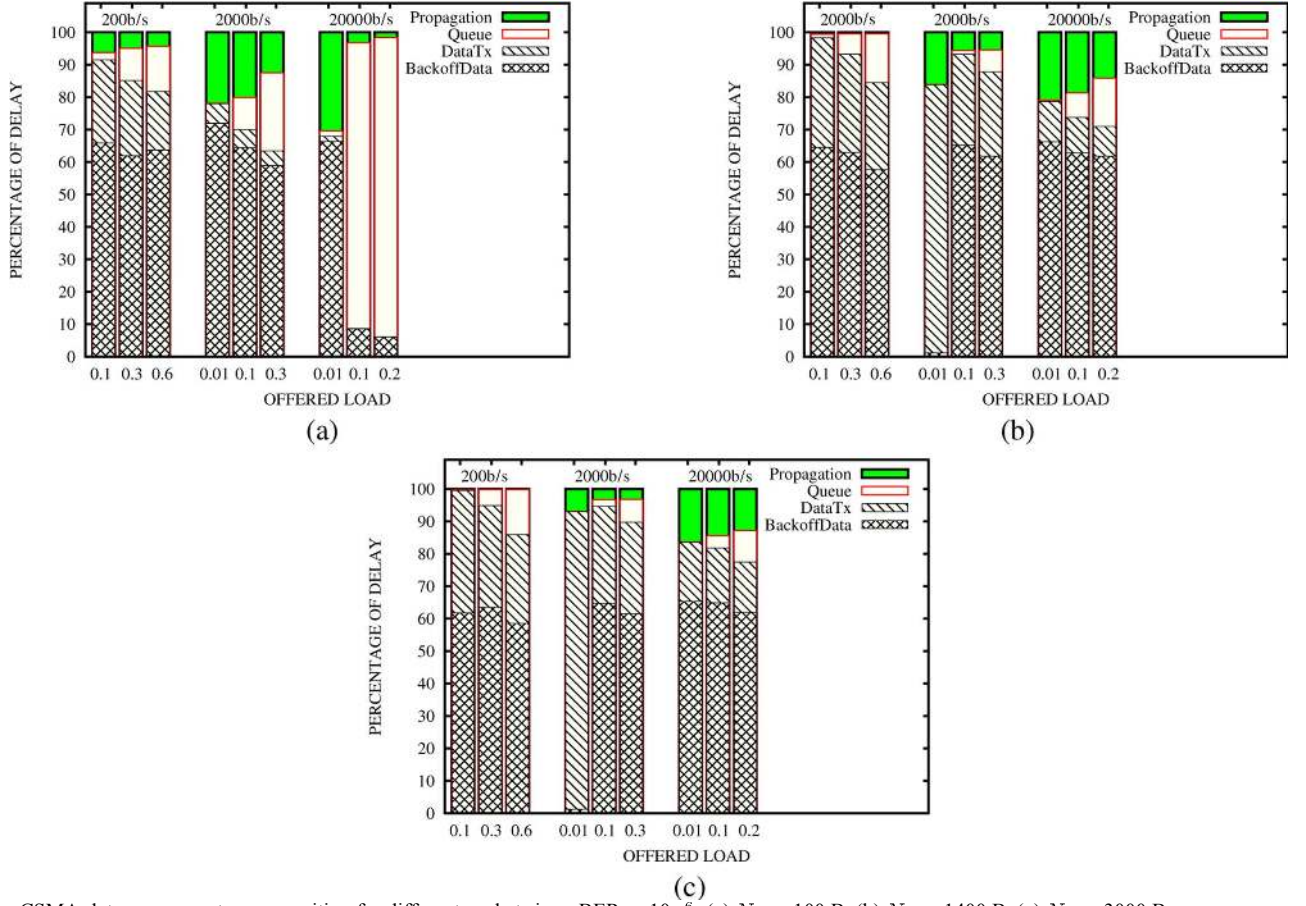


Fig. 6. CSMA: latency per meter composition for different packet sizes, $BER = 10^{-6}$. (a) $N_b = 100$ B. (b) $N_b = 1400$ B. (c) $N_b = 3000$ B.

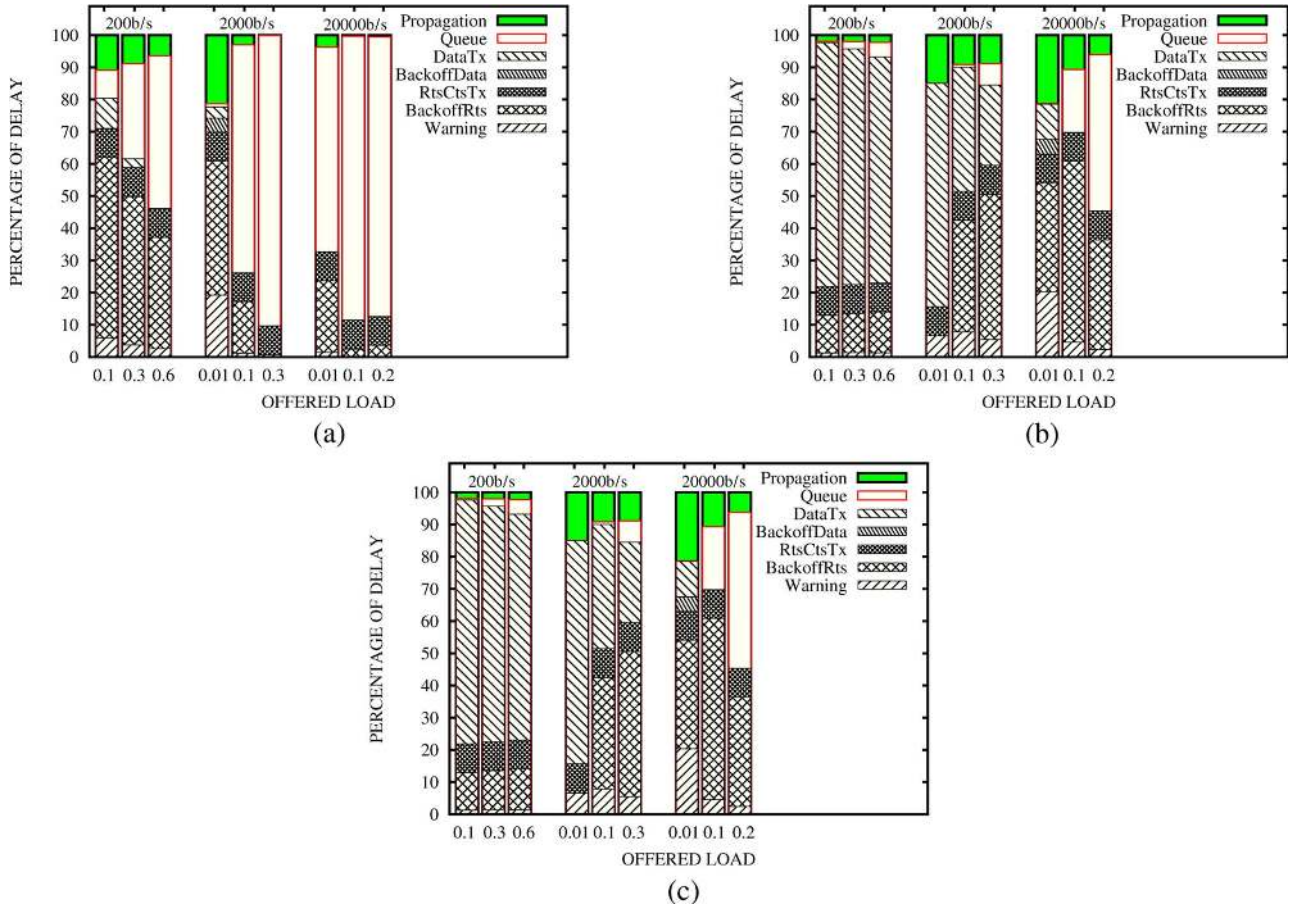


Fig. 7. DACAP: latency per meter composition for different packet sizes, $BER = 10^{-6}$. (a) $N_b = 100$ B. (b) $N_b = 1400$ B. (c) $N_b = 3000$ B.

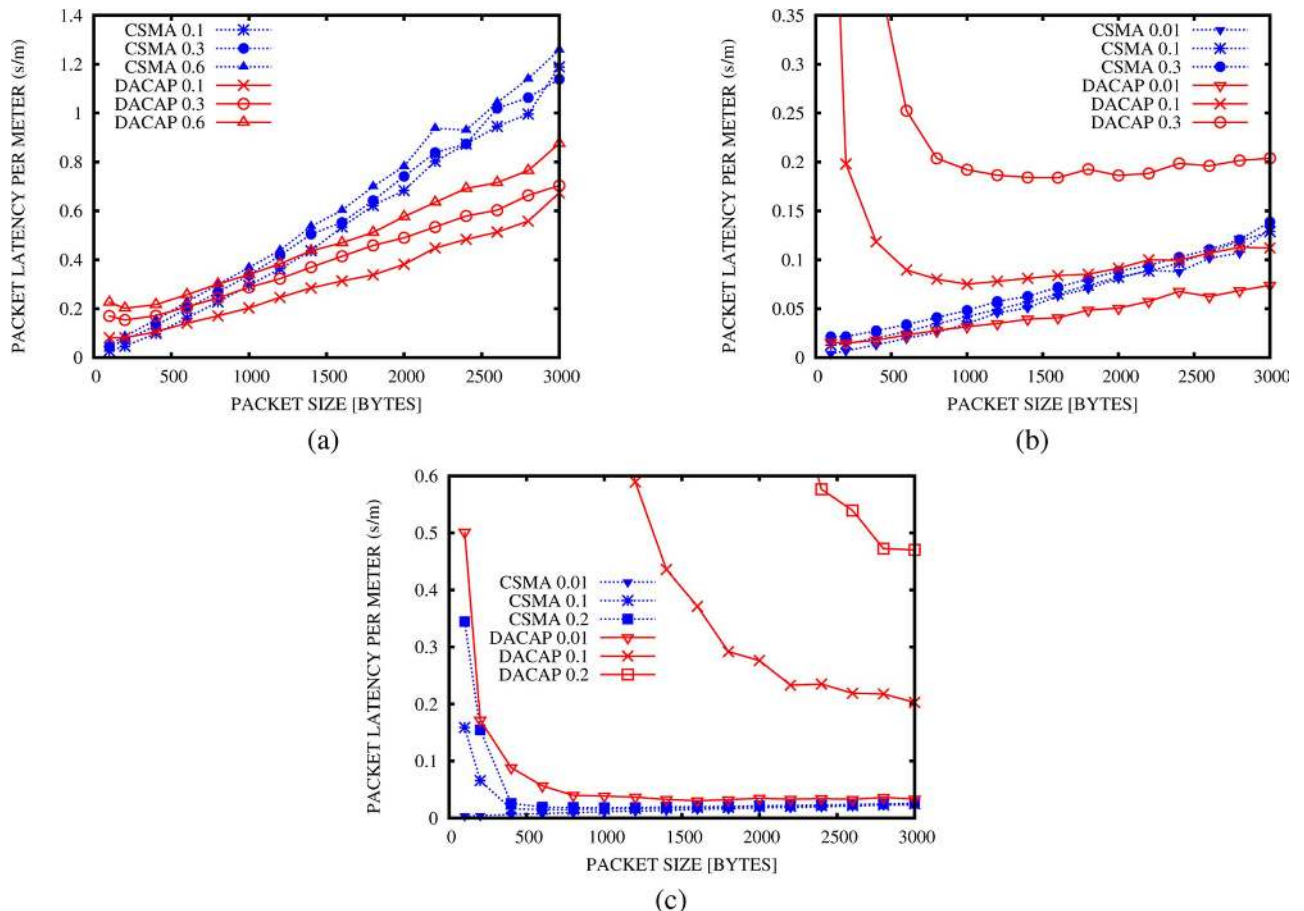


Fig. 8. Latency per meter for different bit rates R_b and BER = 10^{-4} . (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s, zoom. (c) $R_b = 20\,000$ b/s, zoom.

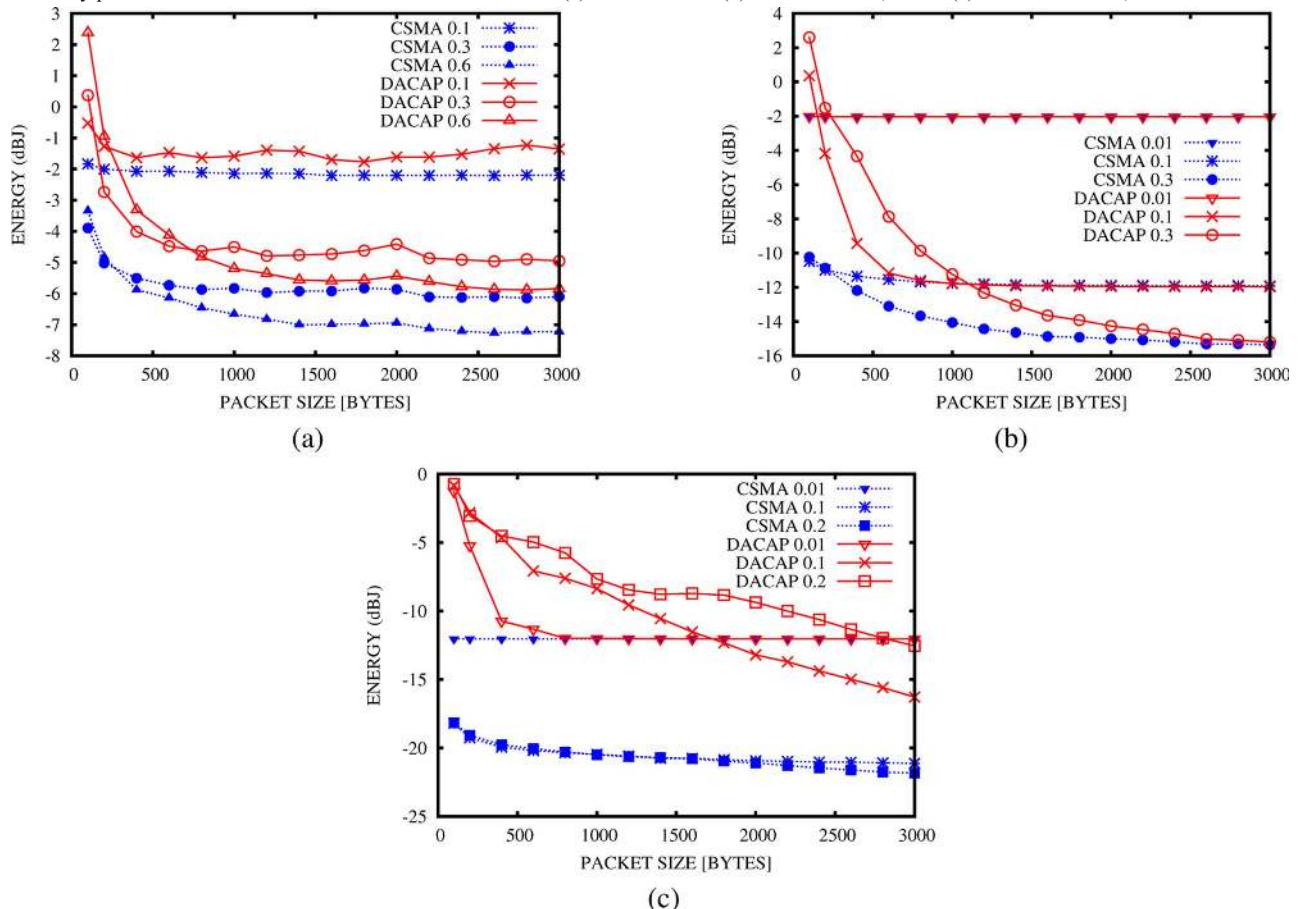


Fig. 9. Energy per bit for different bit rates R_b and BER = 10^{-6} . (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20\,000$ b/s.

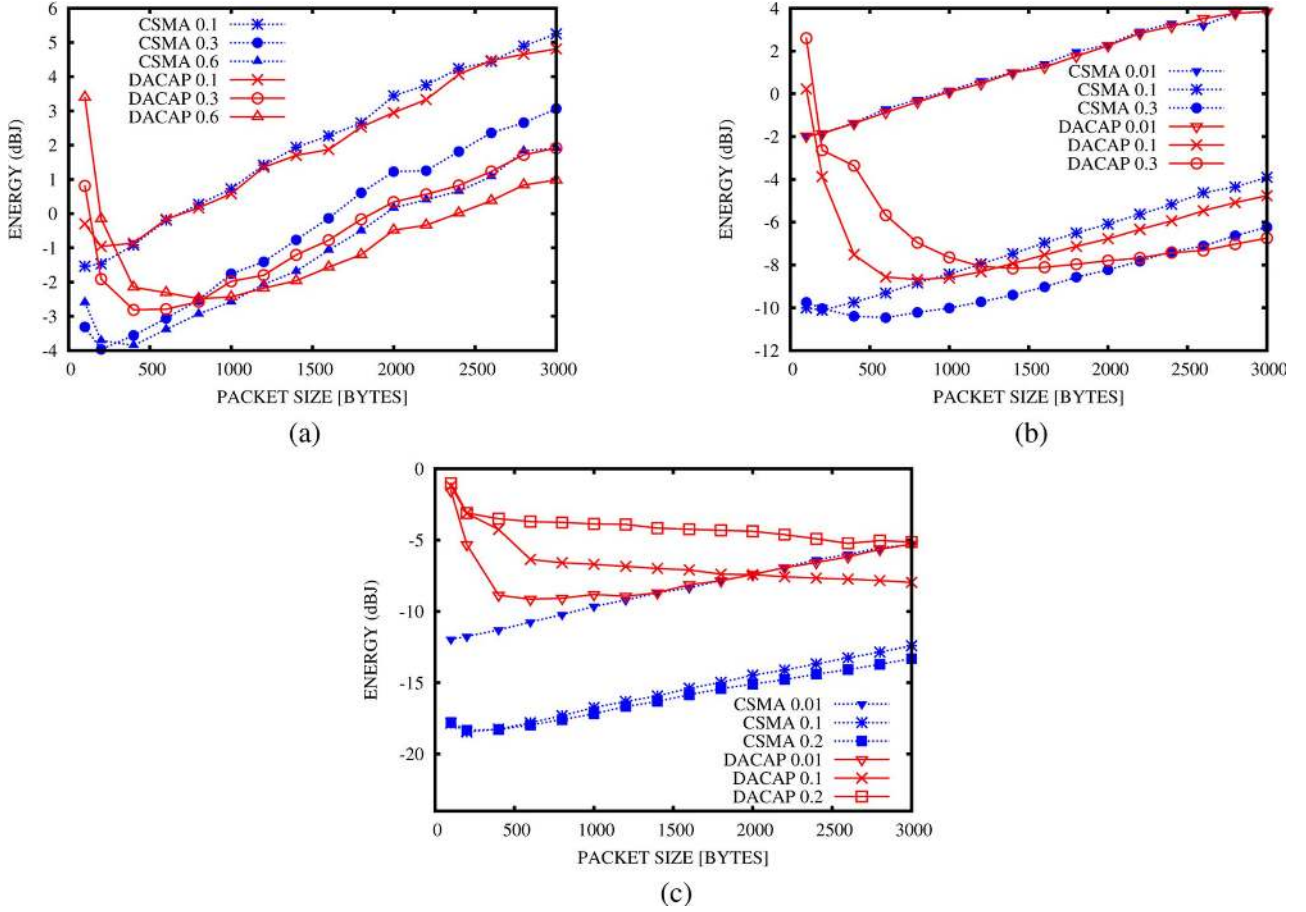


Fig. 10. Energy per bit for different bit rates R_b and $\text{BER} = 10^{-4}$. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20\,000$ b/s.

using very low-power devices to alert a node that relevant communication are upcoming. Considerable advances in this direction are being seen for terrestrial radio nodes [29], [30], and similar developments are ongoing for underwater modems. For instance, Teledyne Benthos modems [31] feature low-power wake up, and Develogic Subsea System Ham.Node [32] implements a very low-power sleep mode as well as a low-power acoustic standby mode. Therefore, we have performed simulations considering nodes equipped with the wake-up capability that could reduce idling and the corresponding energy consumption to negligible values. Results are shown in Figs. 11 and 12 for scenarios with $\text{BER} = 10^{-6}$ and 10^{-4} , respectively.

We observe a remarkable improvement in the energy-per-bit performance of DACAP. Once communication becomes the dominating factor of energy consumption (as opposed to listening), using the RTS/CTS handshaking to limit collisions and retransmissions of long data packets pays off. This is why DACAP shows a performance similar to that of CSMA, and shows better performance for medium/large packet sizes than for shorter packets. We also observe that given $\bar{\lambda}$, as before, delivering more bits correctly results in spending energy more effectively, so that energy-per-bit performance is still related to throughput efficiency. However, increasing the traffic load is no longer beneficial. Even if more bits are delivered correctly, the effectiveness of communication decreases with increasing the load (as retransmissions are needed). When transmissions

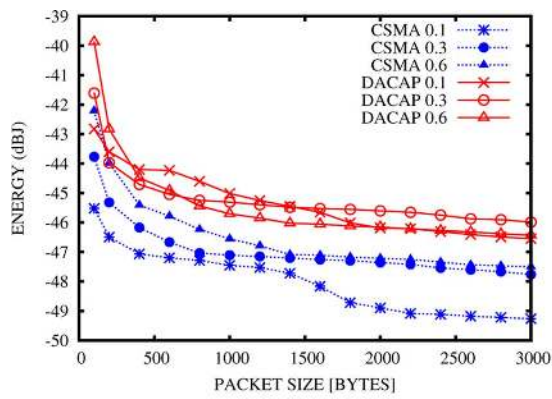
and receptions are the sole factors in energy consumption, a higher load will typically result in decreased energy-per-bit performance.

V. EFFECT OF SYSTEM PARAMETERS ON PACKET SIZE SELECTION

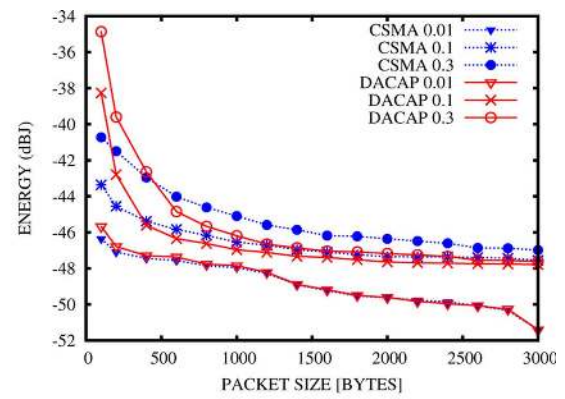
Finally, we investigate how various system parameters affect the optimal packet size. First, we vary parameters such as the network size N , the type of network topology (single-hop versus multihop), and the size of the deployment area, and discuss similarities and differences with the results presented in Section IV. We then show that the inclusion of parameters such as BER, type of protocol, and interference from distant nodes is essential for accurate performance assessment. We also compare our simulation results to those predicted by the high-level analytical model presented by Pompili *et al.* [10], [11].

A. Impact of Varying the Deployment Scenarios

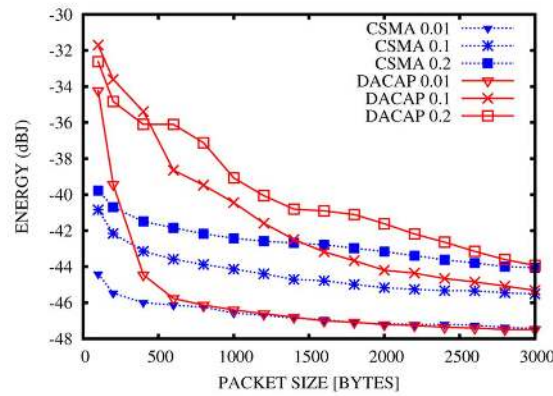
The first set of simulations refers to a network of 15 nodes scattered uniformly in a 700-m \times 700-m area. Each node can transmit directly to the sink, which is located centrally on the surface (single-hop topology). All other parameters are the same as those described in Section III. The packet sizes that optimize throughput performance for both CSMA and DACAP are shown in Fig. 13 for different bit rates R_b and BERs.



(a)

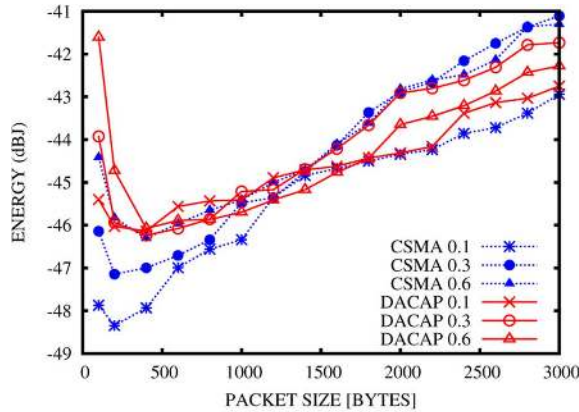


(b)

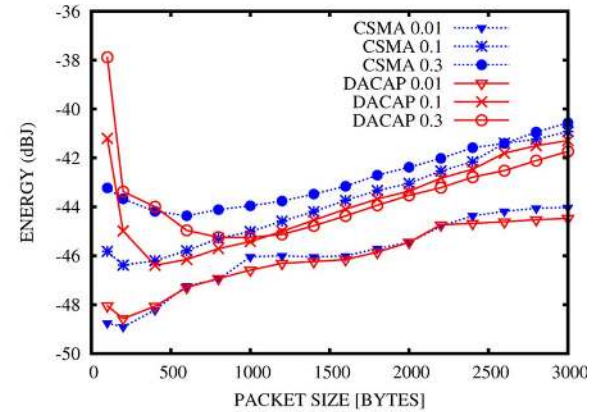


(c)

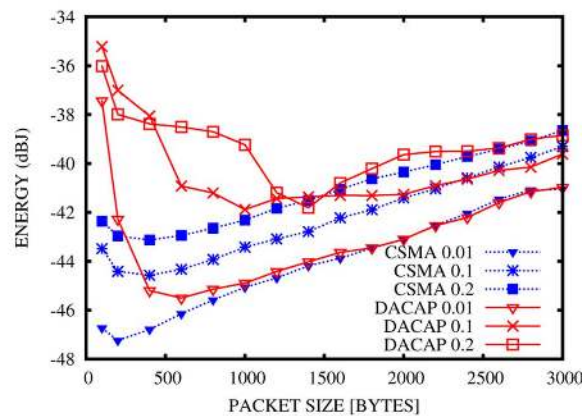
Fig. 11. Scenarios without idle energy consumption: energy per bit for different bit rates R_b and $\text{BER} = 10^{-6}$. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20000$ b/s.



(a)



(b)



(c)

Fig. 12. Scenarios without idle energy consumption: energy per bit for different bit rates R_b and $\text{BER} = 10^{-4}$. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20000$ b/s.

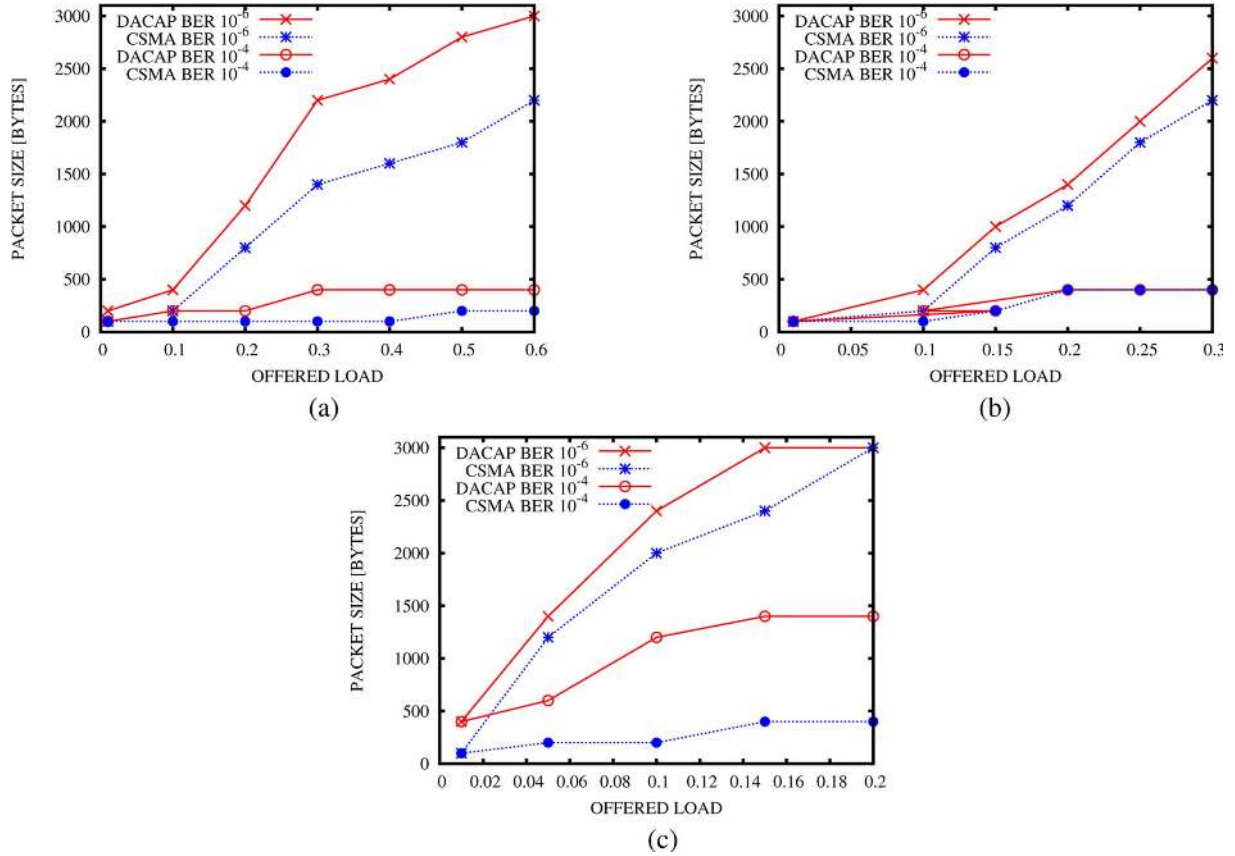


Fig. 13. Packet sizes that optimize throughput efficiency for different bit rates R_b and different BERs for $N = 16$. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20000$ b/s.

A second set of simulations concerns a multihop network with 34 nodes, scattered uniformly in an area of $2000 \text{ m} \times 2000 \text{ m}$. As before, the sink is placed centrally on the surface. The average route length traveled by packets from the nodes to the sink is 1.5 hops. Results for this set of simulations are depicted in Fig. 14.

In general, throughput efficiency, latency per meter, and energy-per-bit consumption show similar trends in all considered scenarios ($N = 16$, $N = 35$, and $N = 100$). When the BER is 10^{-6} , increasing the packet size reduces the overhead, leading to better throughput efficiency. Increasing the network size and the route length increases the network traffic, favoring larger packet sizes for a given offered load. Larger packets are particularly beneficial for DACAP, because of the control overhead required for channel acquisition. In networks with $\text{BER} = 10^{-4}$ we observe two contrasting effects: increasing the packet size reduces the number of contentions; however, at the same time, the higher BER makes it more likely for a larger packet to be discarded because of errors. The combination of these two effects causes the optimal packet size to decrease with the BER.

Despite the fact that trends are similar for different scenarios, the values of packet sizes depicted in Figs. 4, 13, and 14 are noticeably different, suggesting that the packet size needs to be carefully tuned to the specific scenario for optimum performance. Otherwise, the loss on throughput can be significant. For instance, in networks with 16 nodes, bit rate of 20 000 b/s,

$\text{BER} = 10^{-6}$, $\bar{\lambda} = 0.05$, and a packet size of 1400 B, DACAP delivers all packets to the sink. Increasing the number of nodes to 35 (100) makes the throughput efficiency drop to 55% (24%).

B. Comparison Between Simulations and Analytical Models

Packet size optimization has been investigated analytically through the definition and solution of mathematical models [10], [11], [19]. These models have the advantage of providing a general framework; however, to do so, it is necessary to rely on simplifying assumptions, needed for mathematical and computational tractability. To investigate the effect of such simplifying assumptions on the metrics investigated in this paper, we have compared the number of packet retransmissions on a single link as formulated by the analytical model of [11] to that obtained through simulation under CSMA. We have focused our comparison on the expected number of packet retransmissions because this is the core parameter of the model, which determines the dependence of throughput efficiency on the packet size. The comparison, shown in Figs. 15 and 16 for networks with 100 nodes, shows that when the traffic load is low enough not to generate noticeable collisions due to interference, the performance predicted by the analytical model matches well with the results obtained through simulations. However, when the traffic increases, the number of retransmissions obtained via simulation can be up to an order of magnitude higher.

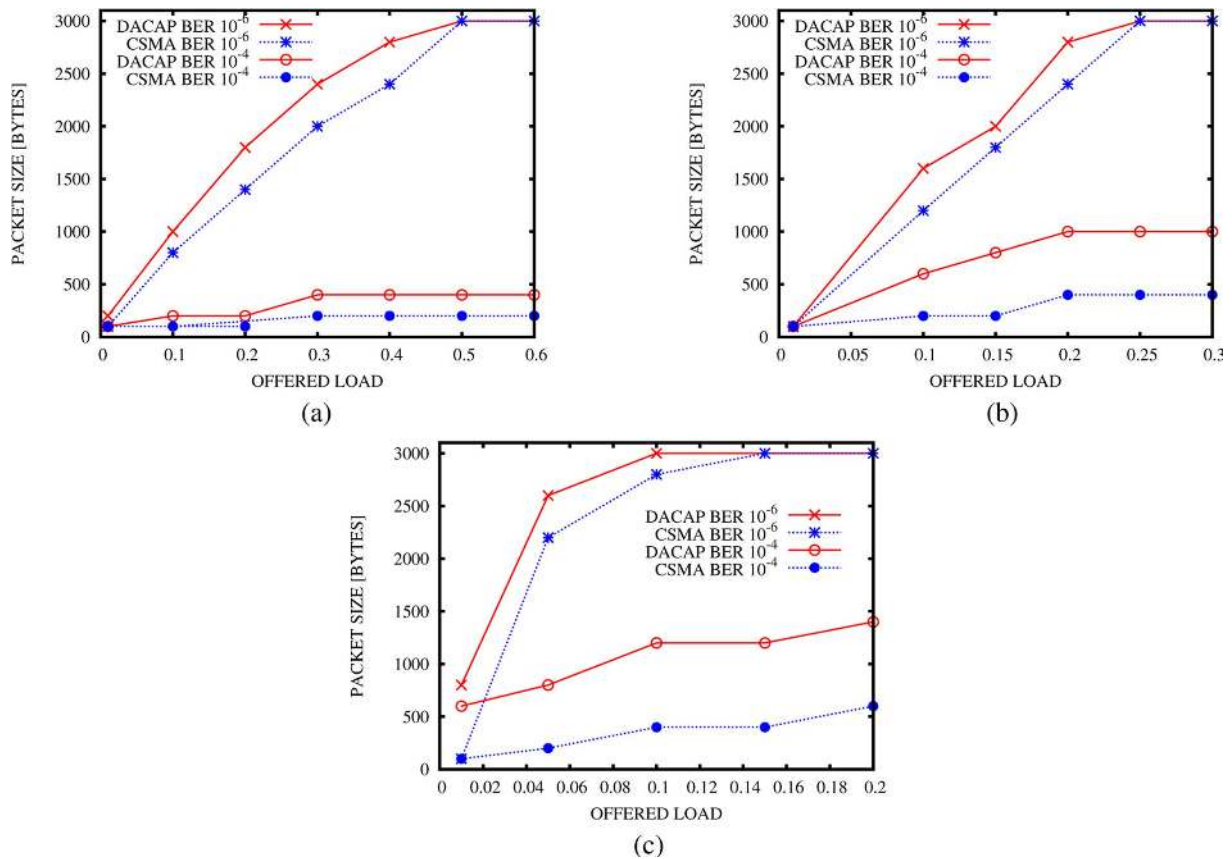


Fig. 14. Packet sizes that optimize throughput efficiency for different bit rates R_b and different BERs for $N = 35$. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20000$ b/s.

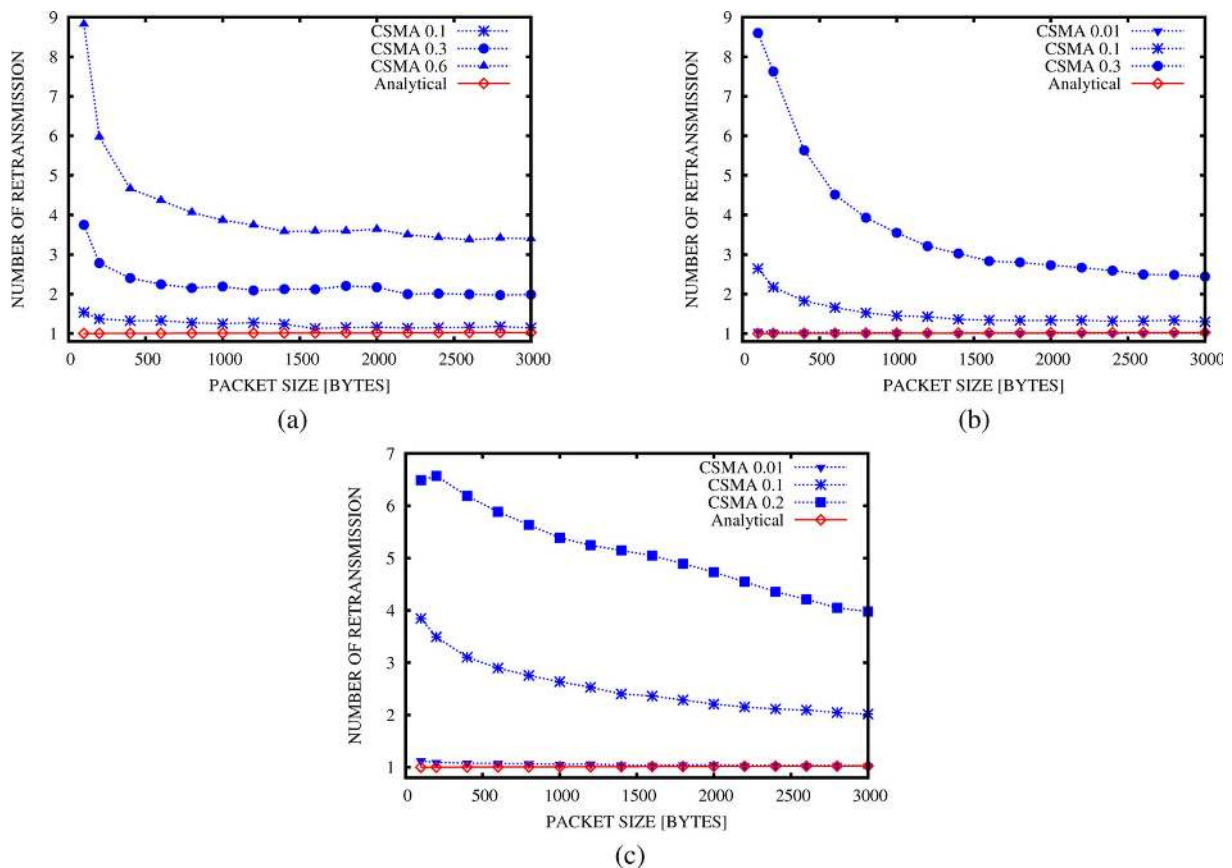


Fig. 15. Average number of data packet retransmission for different bit rates R_b and BER = 10^{-6} . (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20000$ b/s.

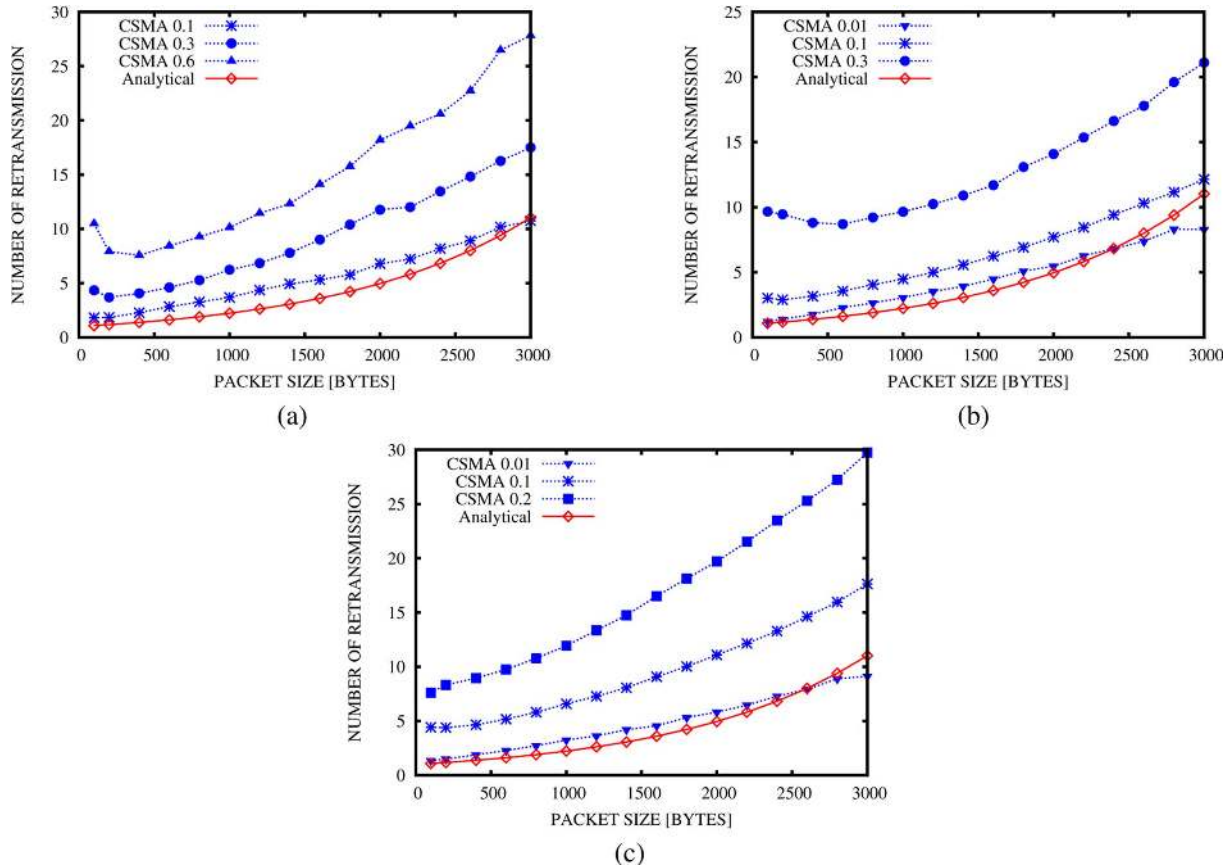


Fig. 16. Average number of data packet retransmission for different bit rates R_b and $\text{BER} = 10^{-4}$. (a) $R_b = 200$ b/s. (b) $R_b = 2000$ b/s. (c) $R_b = 20000$ b/s.

VI. CONCLUSION

The majority of existing acoustic modems are designed to use *a priori* determined packet sizes, which may not be an optimal strategy if a variety of deployment conditions are targeted. To address this issue, we have analyzed the impact of varying the packet size on the performance of an underwater multihop network. We have focused on CSMA and DACAP, two exemplary MAC protocols for underwater networks, and evaluated their performance in light of packet size selection. In doing so, we have allowed for nonzero BER and interference, parameters that were not considered in previous analysis. We observed that appropriate selection greatly depends on the system parameters (bit rate and BER), traffic (packet arrival rate), and the chosen protocol. Results show that CSMA, which does not rely on the extensive usage of control packets, is favorable with shorter data packets, while DACAP, whose collision avoidance is implemented explicitly through a full handshake, shows better performance with long data packets. When network nodes are equipped with low-power wake-up capabilities, we observed benefits to energy consumption, especially for DACAP, whose performance becomes similar to that of CSMA, or better with larger packet sizes. These findings have an important implication on the design of practical acoustic systems, as they point to the fact that choosing a packet size *a priori*, in an *ad hoc* manner, may severely penalize the overall throughput performance.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers whose comments led to improvements of paper presentation and results.

REFERENCES

- [1] I. F. Akyildiz, D. Pompili, and T. Melodia, "State-of-the-art in protocol research for underwater acoustic sensor networks," *SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 11, no. 4, pp. 11–22, Oct. 2007.
- [2] J. Heidemann, W. Ye, J. Willis, A. A. Syed, and Y. Li, "Research challenges and applications for underwater sensor networking," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Las Vegas, NV, Apr. 3–6, 2006, pp. 229–235.
- [3] M. Chitre, S. Shahabudeen, and M. Stojanovic, "Underwater acoustic communications and networking: Recent advances and future challenges," *Mar. Technol. Soc. J.*, vol. 42, no. 1, pp. 103–116, 2008.
- [4] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "ALOHA-based MAC protocols with collision avoidance for underwater acoustic networks," in *Proc. 26th IEEE Int. Conf. Comput. Commun.*, Anchorage, AK, May 6–12, 2007, pp. 2271–2275.
- [5] X. Guo, M. Frater, and M. Ryan, "An adaptive propagation-delay-tolerant MAC protocol for underwater acoustic sensor networks," *IEEE J. Ocean. Eng.*, vol. 34, no. 2, pp. 170–180, Apr. 2009.
- [6] J. M. Jornet, M. Stojanovic, and M. Zorzi, "Focused beam routing protocol for underwater acoustic networks," in *Proc. 3rd ACM Int. Workshop UnderWater Netw.*, San Francisco, CA, Sep. 15, 2008, pp. 75–82.
- [7] R. Nitzel, C. Benton, S. Chappell, and D. Blidberg, "Exploiting dynamic source routing to enable undersea networking over an ad-hoc topology," in *Proc. Int. Symp. Underwater Technol.*, Apr. 2002, pp. 273–277.

- [8] B. Peleato and M. Stojanovic, "Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks," *IEEE Commun. Lett.*, vol. 11, no. 12, pp. 1025–1027, Dec. 2007.
- [9] C. Petrioli, R. Petroccia, and M. Stojanovic, "A comparative performance evaluation of MAC protocols for underwater sensor networks," in *Proc. MTS/IEEE OCEANS Conf.*, Quebec City, QC, Canada, Sep. 15–18, 2008, DOI: 10.1109/OCEANS.2008.5152042.
- [10] D. Pompili, T. Melodia, and I. F. Akyildiz, "A resilient routing algorithm for long-term applications in underwater sensor networks," presented at the IFIP Annu. Mediterranean Ad Hoc Netw. Workshop, L'ipari, Italy, Jun. 14–17, 2006.
- [11] D. Pompili, T. Melodia, and I. F. Akyildiz, "Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks," in *Proc. Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 23–26, 2006, pp. 298–309.
- [12] A. Syed, W. Ye, and J. Heidemann, "Understanding spatio-temporal uncertainty in medium access with ALOHA protocols," in *Proc. 3rd ACM Int. Workshop UnderWater Netw.*, Montréal, QC, Canada, Sep. 14, 2007, pp. 41–48.
- [13] A. Syed, W. Ye, and J. Heidemann, "Comparison and evaluation of the T-Lohi MAC for underwater acoustic sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 9, pp. 1731–1743, Dec. 2008.
- [14] L. T. Tracy and S. Roy, "A reservation MAC protocol for ad-hoc underwater acoustic sensor networks," in *Proc. 3rd ACM Int. Workshop UnderWater Netw.*, San Francisco, CA, Sep. 15, 2008, pp. 95–98.
- [15] M. Zorzi, P. Casari, N. Baldo, and A. Harris, "Energy-efficient routing schemes for underwater acoustic networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 9, pp. 1754–1766, Dec. 2008.
- [16] M. Stojanovic, "Optimization of a data link protocol for an underwater acoustic channel," in *Proc. IEEE OCEANS Europe Conf.*, Brest, France, Jun. 20–23, 2005, vol. 1, pp. 68–73.
- [17] H.-H. Ng, W.-S. Soh, and M. Motani, "MACA-U: A media access protocol for underwater acoustic networks," in *Proc. IEEE Global Telecommun. Conf.*, New Orleans, LA, Nov. 4, 2008, DOI: 10.1109/GLOCOM.2008.ECP.165.
- [18] A. S. Tanenbaum and D. Wetherall, *Computer Networks*, 5th ed. Upper Saddle River, NJ: Prentice-Hall, 2010.
- [19] M. C. Vuran and I. F. Akyildiz, "Cross-layer packet size optimization for wireless terrestrial, underwater, and underground sensor networks," in *Proc. IEEE Conf. Comput. Commun.*, Phoenix, AZ, Apr. 13–18, 2008, pp. 226–230.
- [20] S. Basagni, C. Petrioli, R. Petroccia, and M. Stojanovic, "Multiplexing data and control channels in random access underwater networks," in *Proc. MTS/IEEE OCEANS Conf.*, Biloxi, MS, Oct. 26–29, 2009, pp. 1–7.
- [21] The VINT Project, "The network simulator—ns-2," 2002 [Online]. Available: <http://www.isi.edu/nsnam/ns/>
- [22] M. Porter, "Bellhop code," [Online]. Available: <http://oalib.hlsresearch.com/Rays/index.html>
- [23] 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 *Proc. 2nd Int. Conf. Performance Eval. Methodol. Tools*, Nantes, France, Oct. 23–25, 2007, article 16.
- [24] F. Guerra, P. Casari, and M. Zorzi, "World Ocean Simulation System (WOSS): A simulation tool for underwater networks with realistic propagation modeling," in *Proc. 4th ACM Int. Workshop UnderWater Netw.*, Berkeley, CA, Nov. 3, 2009, DOI: 10.1145/1654130.1654134.
- [25] National Oceanographic Data Center (NODC), "World Ocean Atlas," [Online]. Available: www.nodc.noaa.gov/OC5/WOA05/pr_woa05.html
- [26] GEBCO, "General bathymetric chart of the oceans," [Online]. Available: www.gebco.net
- [27] National Geophysical Data Center, "Seafloor surficial sediment descriptions," [Online]. Available: <http://www.ngdc.noaa.gov/mgg/geology/deck41.html>
- [28] L. Freitag, M. Grund, S. Singh, J. Partan, P. Koski, and K. Ball, "The WHOI Micro-Modem: An acoustic communications and navigation system for multiple platforms," 2005 [Online]. Available: <http://www.whoi.edu/micromodem/>
- [29] L. Gu and J. A. Stankovic, "Radio-triggered wake-up for wireless sensor networks," *Real-Time Syst.*, vol. 29, no. 2–3, pp. 157–182, Mar. 2005.
- [30] G. Lu, D. De, M. Xu, W.-Z. Song, and B. Shirazi, "A wake-on sensor network," in *Proc. 7th ACM Conf. Embedded Netw. Sensor Syst.*, Nov. 4–6, 2009, pp. 341–342.
- [31] Teledyne Benthos, "Undersea systems and equipment," [Online]. Available: <http://www.benthos.com/acoustic-teleonar-modem-product-comparison.asp>
- [32] Develogic Subsea Systems, "Modular hydro acoustic modem—Ham Node," [Online]. Available: <http://www.develogic.de/products/underwater-communication-systems/ham-node/>



Stefano Basagni (S'96–M'03–SM'06) received the Ph.D. degree in computer science from the University of Milano, Milan, Italy, in May 1998 and the Ph.D. degree in electrical engineering from the University of Texas at Dallas, Dallas, in December 2001.

He is currently an Associate Professor in the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA. His current research interests concern research and implementation aspects of mobile networks and wireless communications systems, mobile *ad hoc* and sensor

networking, definition and performance evaluation of network protocols, and theoretical and practical aspects of distributed algorithms. He has published over seven dozen highly cited refereed technical papers and book chapters. He is also coeditor of three books.

Dr. Basagni has served as guest editor of special issues of journals such as Springer's *Mobile Networks and Applications*, Wiley-Interscience's *Wireless Communications and Mobile Computing*, Elsevier's *Ad Hoc Networks*, and Elsevier's *Algorithmica*. He serves as a member of the editorial board and of the technical program committee of several Association for Computing Machinery (ACM) and IEEE journals and international conferences. He is a senior member of the ACM (including the ACM SIGMOBILE), a senior member of the IEEE Computer and Communication Societies, and a member of the American Society for Engineering Education (ASEE) and of the Council on Undergraduate Research (CUR).



Chiara Petrioli (S'96–M'98–SM'06) received the Ph.D. degree in computer engineering from the Università di Roma "La Sapienza," Rome, Italy, in May 1998.

She is currently an Associate Professor in the Computer Science Department, Università di Roma "La Sapienza." Her research interests focus on the design, evaluation, and testing of novel telecommunications systems, focusing on energy-efficient communications, terrestrial and underwater sensor networks, mobile and wireless networks, and the

next-generation Internet. She has published over seven dozen papers in prominent international journals and conferences.

Dr. Petrioli is a member of the steering committee and an Associate Editor of the IEEE TRANSACTIONS ON MOBILE COMPUTING, a member of the steering committee of the ACM Conference on Embedded Networked Sensor Systems (ACM SENSYS), and an Associate Editor of the ACM/Springer *Wireless Networks* journal. She has been a member of the executive committee of ACM SIGMOBILE, and has been Program Co-Chair of leading conferences in the field such as the ACM Annual International Conference on Mobile Computing (ACM MobiCom) and the IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (IEEE SECON). She was a Fulbright scholar.



Roberto Petroccia (M'10) received the Laurea degree in computer science with the highest honors and the Ph.D. degree from the Università di Roma "La Sapienza," Rome, Italy, in 2006 and 2010, respectively.

He is currently a Postdoctoral Researcher at the Università di Roma "La Sapienza." His research interests include sensor networks protocols, underwater communications, and networks. He has published over a dozen papers in international journal and conferences.



Milica Stojanovic (S'90–M'93–SM'08–F'10) graduated from the University of Belgrade, Belgrade, Serbia, in 1988 and received the M.S. and Ph.D. degrees in electrical engineering from Northeastern University, Boston, MA, in 1991 and 1993.

After a number of years with the Massachusetts Institute of Technology (MIT), Cambridge, where she was a Principal Scientist, she joined the faculty of Electrical and Computer Engineering Department, Northeastern University, in 2008. She is also a Guest Investigator at the Woods Hole Oceanographic Institution, Woods Hole, MA, and a Visiting Scientist at MIT. Her research interests include digital communications theory, statistical signal processing and wireless networks, and their applications to underwater acoustic communication systems.

Dr. Stojanovic is an Associate Editor for the IEEE JOURNAL OF OCEANIC ENGINEERING and the IEEE TRANSACTIONS ON SIGNAL PROCESSING.