Game Theory Framework for MAC Parameter Optimization in Energy-Delay Constrained Sensor Networks

A

MESSAOUD DOUDOU, CERIST Research Center, Algiers, Algeria
JOSE M. BARCELO-ORDINAS, Universitat Politecnica de Catalunya (UPC), Barcelona, Spain
DJAMEL DJENOURI, CERIST Research Center, Algiers, Algeria
JORGE GARCIA VIDAL, Universitat Politecnica de Catalunya (UPC), Barcelona, Spain
ABDELMADJID BOUABDALLAH, Université de Technologie de Compiègne (UTC), Compiègne, France
NADJIB BADACHE, University of Sciences and Technology (USTHB), Algiers, Algeria

Optimizing energy consumption and end-to-end (e2e) packet delay in energy-constrained, delay-sensitive wireless sensor networks is a conflicting multi-objective optimization problem. We investigate the problem from a game theory perspective, where the two optimization objectives are considered as game players. The cost model of each player is mapped through a generalized optimization framework onto protocol specific MAC parameters. From the optimization framework, a game is first defined by the Nash Bargaining Solution (NBS) to assure energy-consumption and e2e delay balancing. Secondly, the Kalai-Smorodinsky Bargaining Solution (KSBS) is used to find equal proportion of gain between players. Both methods offer a bargaining solution to the duty-cycle MAC protocol under different axioms. As a result, given the two performance requirements, i.e., the maximum latency tolerated by the application and the initial energy budget of nodes, the proposed framework allows to set tunable system parameters to reach a fair equilibrium point which dually minimizes the system latency and energy consumption. For illustration, this formulation is applied to six state-of-the-art Wireless Sensor Network (WSN) MAC protocols; B-MAC, X-MAC, RI-MAC, SMAC, DMAC, and LMAC. The paper shows the effectiveness and scalability of such framework in optimizing protocol parameters that achieve a fair energy-delay performance trade-off under the application requirements.

1. INTRODUCTION

Maximizing the network lifetime while assuring the application requirements in terms of end-to-end (e2e) delay is challenging in distributed energy-constrained wireless networks such as Wireless Sensor Networks (WSN), where there is an inherent conflict between the design consideration of the two performance goals. The MAC layer plays a pivotal role in determining the system performance in terms of packet delay and power-consumption (network lifetime), as it manages the radio which consumes the largest amount of a node's battery [Doudou et al. 2013]. Energy saving is achieved at the MAC protocol by duty-cycling the radio and repeatedly switching it between active and sleep modes. In active mode, a node can receive and transmit packets, while in the sleep mode, it completely turns off its radio to save energy. However, forwarding a packet over multiple hops in duty-cycled MAC protocols often requires multiple

---

operational cycles, where nodes have to wait for the next cycle to forward data at each hop. Given the application constraints/requirements in terms of initial energy budget devoted to each individual node and the maximum e2e packet delay tolerated, the choice of MAC protocol's parameters is of high importance; yet their choice is currently determined by system designers based on repeated real experiences [Ceriotti et al. 2011], or on optimizing one objective subject to other objectives as constraints [Zimmerling et al. 2012], [Park et al. 2011].

In this paper, we investigate the inherent trade-off between energy consumption and e2e delay from a game theory perspective. The main result of this work is then to provide a framework that given two well-known system performance requirements, such as the maximum latency tolerated by the application and the initial energy budget devoted to nodes' batteries, it enables the system designer[1] to obtain tunable system parameters that dually minimize latency and energy consumption. In the proposed framework, the players are the performance metrics (energy and delay) instead of the individual nodes that are common in state-of-the-art models that use game-theory for optimizing wireless network MAC protocols. The cost function of each player is used by the game theory optimization framework to determine the MAC protocol optimal parameters. Each player threats the other with using his best optimal point obtained from a non-cooperative game in which the player finds his best optimal operating point, i.e., player delay obtains its lowest delay at the cost of increasing energy consumption, while player energy obtains its lowest energy consumption at the cost of increasing delay. A bargaining game is then defined in order to find an agreement operational point that satisfies both players. Two bargaining solutions are met: the Nash Bargaining Solution (NBS) and the Kalai-Smorodinsky Bargaining Solution (KSBS). The choice of the NBS model or the KSBS model as solution is axiomatic dependent. The KSBS model has the advantage of equal proportion of gain between players, and it has then a property of fairness between the two performance metrics. The fairness property has its applicability in those cases in which the network designer has no preference among the two metrics, while at the same time both application objectives are met. Due to the difficulty of formulating the KSBS optimization problem in its convex form, we propose an algorithm based on iterative NBS resolution, which provides an approximative KSBS solution.

The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3 introduces the general optimization model and the cooperative game-theory framework. Section 4 shows how to derive the energy-delay trade-off for the RI-MAC protocol. The optimization results and the NBS and KSBS models are given in this section with some discussions for six illustrative energy-delay efficient MAC protocols: B-MAC, X-MAC, RI-MAC, SMAC, DMAC, and LMAC. All these protocols and their optimization models, except RI-MAC, are described in Appendix A. In Section 5, we validate the results obtained by the proposed approach through extensive simulations. Finally, Section 6 concludes the paper.

2. RELATED WORK

Performance optimization through protocol modeling is an appealing way to achieve the desired design considerations for any application. While most of the energy-efficient MAC protocol works for wireless networks followed pure experimental approaches, some works (that are the most relevant to this one) have attempted to model and analyze the protocols. [Langendoen and Meier 2010] consider traffic and network models for very low data rate applications and analyze energy consumption and average latency of well known MAC protocols. Markov models have been developed to evaluate the energy consumption of some MAC protocols, such as SMAC [Yang and Heinzelman 2009], and DMAC [Zheng et al. 2011]. Formal optimization for energy minimization of SCP-MAC protocol has been investigated by [Ye et al. 2006]. While most models consider single-objective optimization, protocol optimization under application needs in terms of both energy and e2e delay have been considered in [Park et al. 2011] and [Zimmerling et al. 2012]. However, their approaches are based on optimizing energy subject to constraints on the delay.

Game theory is a practical way to solve many network optimization problems. Different game theoretical models including cooperative, non-cooperative, bayesian, differential, evolutionary games, etc. have been explored to deal with common problems in wireless and communication networks [Han et al. 2011]. Resource allocation and band-width sharing have been addressed by many works using game-theoretical approaches such as [Kim et al. 2012] and [Pandremmenou et al. 2013]. Route selection problems have been studied by [Ortn et al. 2013], where three games with different levels of complexity have been proposed as a distributed self-configuring solution in multi-hop wireless systems. [Chu and Sethu 2015] proposes a cooperative game theoretical topology control solution that allows a node to choose the set of neighbors with which it communicates directly, while preserving global goals such as connectivity or

---

[1]The network designer has to know as a prior the sampling frequency and the network topology and should be able to characterize its MAC protocol energy consumption function and the delay from any node to the sink.

coverage. Game theory has also been used to address energy-efficiency and security problems in wireless sensor networks [Machado and Tekinay 2008]. [Hayel et al. 2014] proposes a decentralized optimal protection non-cooperative game against virus propagation over a network through a Susceptible Infected Susceptible (SIS) epidemic process. The performance of the equilibria has been evaluated by finding the Price of Anarchy (PoA) in several network topologies. Energy-efficiency is also addressed by [Voulkidis et al. 2013], using game-theory-based coalition formation between spatial correlated sensors that reduces the amount of transmitted packets. [Mihaylov et al. 2011] present a game-theory mechanism to save node's energy by playing the *win-stay lose-shift* (WSLS) game between nodes to schedule radio transmissions. [AlSkaif et al. 2015] gives a taxonomy of games applied to wireless sensor networks.

There is a wide literature that study trade-offs between metrics in networking applying different approaches. One of the most investigated examples is the trade-off of delay-throughput in mobile networks, [Gamal et al. 2004][Neely and Modiano 2005], among others. In these works, mostly based on the Gupta-Kumar fixed traffic model [Gupta and Kumar 2000] or on the Grossglauser-Tse mobile model [Grossglauser and Tse 2001], the trade-off between throughput or capacity and delay is investigated under several scenarios such as the transmission range, interference, number of hops, degree of mobility and draw throughput-delay scaling trade-off curves. [Ying et al. 2008], first characterize the maximum throughput per source-destination and then develop a joint coding-scheduling scheme for finding the delay-throughput optimal trade-off. Other examples of finding trade-offs are related to spectrum sharing in cognitive networks. The energy-delay tradeoff have been considered by [Neely 2007] in multi-user wireless downlink and by [Leow and Pishro-Nik 2007] in sensor networks using stochastic optimal networking model. [Zhang and Tang 2013] studied the Power-Delay Tradeoff over OFDMA Networks and proposed a resource allocation schemes to minimize the power consumption subject to a delay quality-of-service (QoS) constraint, expressed in terms of queue-length. [Zhang et al. 2012] investigate the energy-delay tradeoff in a wireless multihop network with unreliable link model and provide a solution based on single-objective optimization method for different channel models (additive white Gaussian noise, Rayleigh fast fading and Rayleigh block-fading). From game-theory perspective, [Afghah et al. 2013] proposes a Stackelberg game formulation to address spectrum allocation and includes fairness and energy-efficiency in the game definition. In the same area, [Niyato and Hossain 2008] investigate spectrum trading and propose several pricing models in cognitive radio environments. In our case, our approach to achieve fairness is based on the two-person bargaining problem as a way of obtaining a Pareto efficient point in which the two players may find an agreement with equal proportion of gain.

Numerous efforts have been devoted to address MAC optimization in wireless networks using game theory. [Nuggehalli et al. 2008] use game-theory approaches to address the QoS support in 802.11 networks that enables users with high-priority (HP) or low-priority (LP) traffic to fairly negotiate channel access. Similar approaches have been introduced by [Bacci et al. 2013], [Ghasemi and Faez 2008], [Meshkati et al. 2009a; Meshkati et al. 2009b] to limit the selfish behavior in using the medium resource by end users controlled by the local power level, respectively in OFDMA systems, CDMA networks, and wireless networks. [Zhao et al. 2009] propose to use a cooperative game to control the contention window CW size of every node, which permits energy saving by estimating the number of competing nodes. [Abrardo et al. 2013] propose a a non-cooperative duty-cycle control game to reduce idle-listening time in asynchronous MAC protocols. The energy-delay trade-off is considered by [Nahir and Orda 2007], where multiple cost models (power level, direct/indirect transmissions) are used by each node to determine the Nash equilibrium point for different communication tasks (unicast, multicast, and broadcast) in wireless networks.

All the above mentioned works consider nodes as players in the game and attempt to maximize the defined utility function. In cooperative games, the coalition may be achieved by players' coordination through messages exchange and thus the complexity of the solution increases when the number of players involved in the competition scale-up. [Zhao et al. 2013] propose a game-theoretical solution where the performance goals are considered as peers (players) of the game. This is interesting since considering the cooperative game between the two system performance objectives makes the complexity of the solution independent from the number of nodes. Although the authors do not deal with WSN but consider the trade-off between load-balancing and energy-efficiency for traffic engineering in communication networks, the work is the most related to the one presented in this paper in terms of modeling and share the way the game players are represented. The proposed solution differs from that of Zhao et al. in the way to achieve the fair trade-off point. In fact, Zhao et al. focus on the NBS model and propose an iterative approach to reach the fair trade-off point which updates the threat values of players by halving on the line that connects (X_{worst}, Y_{worst}) with the solution of the NBS game at each iteration. The accuracy of their algorithm depends on how close is the NBS point from the KSBS point, something that depends on the problem and may not converge at all. Instead, we focus on the fairness using
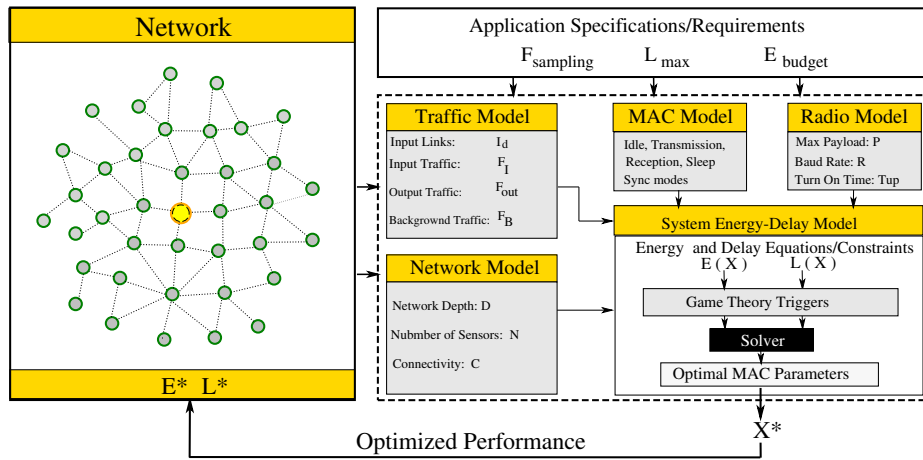
Fig. 1.   Game Theory Optimization Framework

Kalai and Smorodinsky theory and we propose an algorithm to approximate the KSBS point from the NBS solution by adequately updating the threat values so that the solution converges to the KSBS point (the fair trade-off).

The optimization framework proposed in this paper allows to achieve a fair trade-off between energy and delay performance for duty-cycled MAC protocols in WSN. Tunable MAC parameters that enable to achieve this trade-off are accordingly determined. The proposed framework is based on the energy model derived in [Langendoen and Meier 2010], and it uses the Bargaining solution to efficiently balance objectives modeled as virtual players, which is inspired by the model proposed in [Zhao et al. 2013]. Another feature of the proposed framework is fairness. In fact, the notion of fairness was defined by Kelly [Kelly 1997] to allocate resources based on the users rate requirements. From the game theory point of view, the fairness was introduced into the bargaining model by Kalai and Smorodinsky [Kalai and Smorodinsky 1975]. The Kalai and Smorodinsky bargaining (KSBS) solution keeps the the same axioms required by the Nash bargaining solution (NBS), and defines an equity constraint linking the utility gains of each player to their maximum achievable utilities (known as claims point) by equalising proportional gains of individual utilities. The KSBS model was explored by many works to address fair multimedia resources allocation and efficient bandwidth sharing in network traffic management [Park and van der Schaar 2007; Mazumdar et al. 1991], wireless networks [Chen and Swindlehurst 2012; Bastani et al. 2012], and lately applied to sensor networks [Pandremmenou et al. 2013; Ma et al. 2011] for resources allocation and [Truong et al. 2010] for coverage efficiency. Instead, we propose in this paper to achieve the fair solution of KSBS through iterative NBS applied to low data rate duty-cycled sensor networks. To the best of our knowledge, this work is the first that considers the energy/delay trade-off in duty-cycling MAC protocols using game-theory.

3. GAME THEORY FRAMEWORK

In the proposed framework, the key performance metrics are the energy consumption, E, and the maximum e2e packet delay, L. The application requirements expressed as the maximum energy budget per node, E_{budget}, and the maximum allowed e2e delay per packet, L_{max}, are used as inputs for the framework. The framework, Fig. 1, builds then a system model for energy and delay based on (i) the network and traffic models that permit to determine the topology information and the traffic load at each node, and (ii) the specified MAC model defined by its operating modes: idle, transmission, reception, sleep modes, etc. The network designer runs the optimization and game theoretical framework and obtains the optimal values that allow a network deployment that fulfils the delay and energy consumption requirements.

3.1. Network and Traffic Model

Let us consider an unsaturated network with low traffic, which is typical in energy-constrained networks, e.g., WSN applications. A typical network model is considered, following the same analysis as in [Langendoen and Meier 2010]
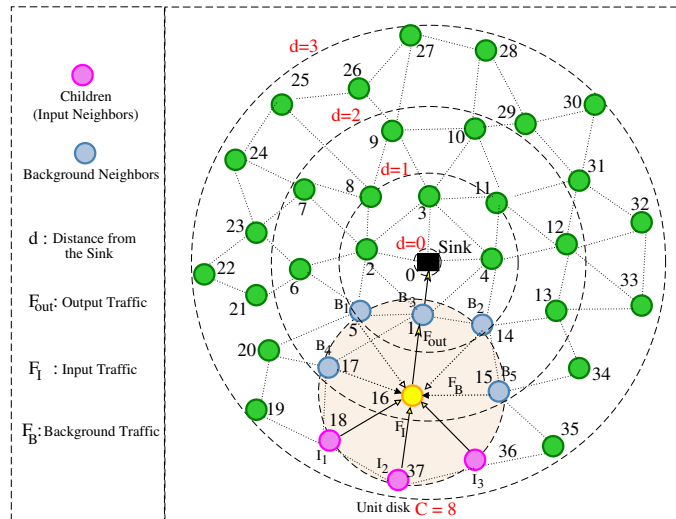
Fig. 2. Typical Network Topology with Local Node Traffic Flowing.

where the authors assume an arbitrary topology layered into rings with the sink node located at the center [2]. A spanning tree is constructed, where nodes are static and maintain a unique path to the sink, and they use the shortest path routing with a maximum length of D hops, i.e., the depth or number of rings of the tree. We assume a network of N nodes, a uniform node density on the plane, and a unit disk graph communication model with density (average number of nodes), C, i.e., unit disks contain $C+1$ nodes (on average). The nodes are layered into levels according to their distance to the sink in terms of minimal hop count, d ($d=1,...,D$), where $d=0$ is reserved for the sink. Periodic traffic generation is considered, where every source node generates traffic with frequency F_s. Consequently, the same input F_I^d, output F_{out}^d, background F_B^d traffic, and input links I^d equations for nodes at ring d are similar to those derived in [Langendoen and Meier 2010]. The different symbols introduced in the analysis related to network and traffic model are summarized in Table I with typical values. The neighboring nodes can be then classified as the set of children (input) nodes, I, and the set of overheard (background) nodes, B, such that, $C = |I| + |B|$. Each node at the same level, d, has on average the same input, output, and background traffic. Thus, the average input traffic, F_I^n, output traffic, F_{out}^n, and background traffic, F_B^n, for a specific node n at level d are respectively defined as: $F_I^n = F_I^d|_{n \in d}$, $F_{out}^n = F_{out}^d|_{n \in d}$, and $F_B^n = F_B^d|_{n \in d}$.

3.2. Radio and MAC Model

The optimization framework requires information about the radio hardware conjointly with the MAC layer model in order to build the system energy and delay model. The packet-based radio model is consider in our framework. The reason behind this choice is twofold: first, this model is used by most of the well known hardware platforms, such as MicaZ and TelosB, which both use CC2420 [CC2 2010]. Second, most of the current MAC implementations support this kind of radio that decouples the radio functionality from the link layer. For example, the current implementation of LPL (Low Power Listening) MAC [Moss and Levis 2008], which has been originally designed for bit-stream radios, supports packet-based radio in the recent versions of TinyOS (Sensors operation system [Levis et al. 2004]). These facts lead us to adopt the radio model of CC2420 (packet-based) as defined in [Langendoen and Meier 2010], which models the time needed to power the radio up (i.e., to transit from sleep into active mode), the radio data rate and the time needed for carrier sense (including power-up). The crystal frequency tolerance is another modeled hardware parameter. Due to the low data rate, the clocks of different nodes may drift apart and need to be accounted for. This

---

[2]Other topologies such as grid or with the sink at different locations may also be considered. The framework defined by [Langendoen and Meier 2010] will give other average values for the input/output rates at each node that will impact the delay or energy consumption formulas, but the game theory framework can be easily adapted to these cases.
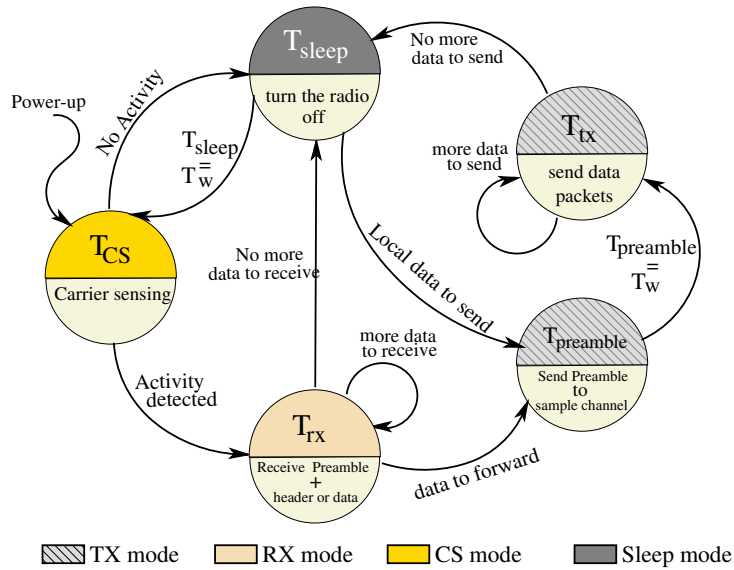
Fig. 3. States transition of different LPL operation modes.

parameter is very important in MAC protocols which permits to compensate for eventual clock drift. The main modeled radio parameters used in this work are summarized in table I with typical values.

Many internal MAC layer parameters are related to the MAC protocol operation. The understanding of the MAC protocol behavior and operation mode is of pivotal importance to determine the role of each parameter. Since the objective of this paper is to optimize the MAC parameters that lead to a fair trade-off between energy and delay performances, the main parameters that affect both performances, such as the wake-up period T_w, the slot period, or the frame size T_{frame}, can be captured, while other details can be left out. By analyzing the operating modes of the MAC protocol, the parameters involved in the delay and energy consumption when the radio is switched between different states (idle listening, transmitting, receiving, and sleeping) can be identified. For example, consider the basic operation modes of LPL [Hill and Culler 2002] protocol depicted in Fig. 3. Each node, when powered-up, performs carrier sensing every wake-up period, T_w. A node is likely to sleep most of the time to save energy and only wakes-up if it has data to send or to check the channel for eventual incoming data packets. If no activity is detected, a node returns back to the sleep state, otherwise it keeps the radio in the receive state until the data packet header is being detected to determine whether to receive the data. When a node has data packets to be sent, it first samples the channel with a preamble that spans for the whole wake-up period, T_w, to ensure that its receiver can hear the preamble. It can be easily determined that the wake-up period, T_w, is the key parameter that affects both energy and delay performances. The same principal is followed through the paper where six MAC protocols are modeled according to their operation modes, and a vector of key tunable parameters that can be optimized are determined and used to derive the energy consumption and the e2e (end to end) packet delay function of a node. The considered MAC model is mainly based on the analysis provided in [Langendoen and Meier 2010] for low data rate applications and extended for other protocols. We keep the same assumptions regarding the absence of interferences and the low rate constraints, and we provide for each protocol the per-node energy consumption based on the protocol operation modes, the e2e packet delay, and the bottleneck constraint, which can be used as input for the optimization framework. An overview of these six protocols and their internal parameters is provided in Table II and Table VI for reference.

3.3. System Energy and Delay Model

The energy consumption of node n, denoted, E^n, is defined as the amount of energy consumed by the radio duty of the node in the network according to its position and the amount of traffic it handles. Thus, the node's energy

Table I. CC2420 Radio Constants [CC2 2010], Network and Traffic Model with Typical Parameter Values.

CC2420 Radio	Parameter Description	Values
R	Rate [kbyte/s]	31.25
θ	Frequency tolerance [ppm]	30
T_{cs}	Time [ms] to turn the radio on and probe the channel (carrier sense)	2.60
T_{up}	Time [ms] to turn the radio on into RX or TX	2.40
L_{pbl}	Packet preamble length [byte]	4
Traffic & Network	Parameter Description	Values
P	Data payload [bytes]	32
F_s	Sampling rate [pkts/node/min]	$[1/60, 2]\ pkts/min$
F_{out}^d	Node's output traffic frequency at level d	$F_s \frac{D^2 - d^2 + 2d - 1}{2d - 1}$
F_I^d	Node's input traffic frequency at level d	$F_{out}^d - F_s$
F_B^d	Background Node's traffic frequency at level d	$|B^d| F_{out}^d$
F_{out}^n	Output traffic frequency at node n	$F_{out}^n = F_{out}^d|_{n \in d}$
F_I^n	Input traffic frequency at node n	$F_I^n = F_I^d|_{n \in d}$
F_B^n	Background traffic frequency at node n	$F_B^n = F_B^d|_{n \in d}$
D	Network depth [#levels]	$[5, 12]$
C	Network density (connectivity) [#neighbors]	8
N	Network size (number of nodes) [#nodes]	$D^2 C$

consumption [3] is the sum of energy consumed in each operating mode, which depends on the exchanged traffic load and the MAC intrinsic parameters. For example, let E_{idle}^n, E_{tx}^n, and E_{rx}^n be the energy consumed fractions for node n in idle listening [4], transmitting and receiving modes, respectively. The node's energy consumption can be calculated as, $E^n = E_{idle}^n + E_{tx}^n + E_{rx}^n$. The normalized energy consumption (in Joules) can be calculated by multiplying the obtained expressions in each mode by the current draws of the radio defined in the datasheet for each mode (e.g., I_{idle}, I_{tx} and I_{rx}). In general, for any MAC protocol in the literature, the node's consumed energy is caused by carrier sensing E_{cs}, data transmission E_{tx}, data receiving E_{rx}, overhearing E_{ovr}, and by sending/receiving explicit synchronization, respectively denoted by E_{stx} and E_{srx}. Given that the network lifetime can be expressed as the expected shortest node-lifetime [Zimmerling et al. 2012], we define the system wide energy consumption, E, as the maximum consumed energy in the network,

$$E = \max_{n \in N}(E^n) = \max_{n \in N}(E_{cs}^n + E_{tx}^n + E_{rx}^n + E_{ovr}^n + E_{stx}^n + E_{srx}^n). \tag{1}$$

More specific MAC protocols will present other energy consumption terms that should be added to eq.(1).

The end-to-end (e2e) packet delay (latency), L^n, is defined as the expected time between the first transmission of a packet at node, $n \in N$, and its reception at the sink. It is then a per-topology parameter, in the sense that it depends on the position of the node that generates the data. L^n denotes the sum of per-hop latencies of the shortest path, $\mathcal{P}^n$, from node n to the sink, where L_l^n is the one-hop latency on each link $l \in \mathcal{P}^n$. The maximum end-to-end latency, L, is defined as the maximum latency from all nodes to the sink as follows:

$$L = \max_{n \in N}(L^n) = \max_{n \in N}\left(\sum_{l \in \mathcal{P}^n} L_l^n\right) \tag{2}$$

3.4. The Optimization Framework

Let Θ denote the set of system parameters at MAC layer that appear in the energy and delay functions, such as sleep length, preambles, packet sizes, etc. Some of these parameters, for example the MAC preamble, are not tunable by the network designer. However, other parameters such as the duty cycle of the sensor may be fixed by the designer. It is obvious that a long duty cycle reduces the latency but causes more energy consumption at idle state. Given a specific

---

[3]The term energy consumption used in this paper refers to the duty-cycle of the radio during the node's lifetime, measured in the percentage of active periods.

[4]The term idle listening is not explicitly mentioned in the equations. It is modeled under different names that differs from a protocol to another. For example, terms T_{cs} and T_{up} in Table I that are common for all MAC protocols are fractions of time where the node is in idle listening and the radio is consuming energy.

duty-cycle MAC protocol, let $X \in \Theta$ be the vector of system parameters that may be tuned and thus optimized. The following optimization problem is defined for energy consumption minimization:

$$\text{(P1)} \quad \min \; E(X)$$
$$\text{s.t.} \quad L(X) \leqslant L_{\max}$$
$$\text{var.} \quad X,$$

where $L_{\max}$ is the maximum latency tolerated by the application. On the other hand, the following optimization problem is defined for latency minimization:

$$\text{(P2)} \quad \min \; L(X)$$
$$\text{s.t.} \quad E(X) \leqslant E_{\text{budget}}$$
$$\text{var.} \quad X,$$

where E_{budget} is the energy budget (maximum duty cycle) that a node consumption cannot exceed for fulfilling a lifetime target. Let us take without loss of generality as example a simple duty-cycled MAC protocol with the length of the wake-up period, T_w, as the single parameter to be optimized, i.e. $X=T_w$. Prolonging this time enables energy saving, since the energy consumption is proportional to the duty cycle and thus inversely proportional to the length of the wake-up period (e.g., E$\sim 1/T_w$). However, this causes high latency for relaying data to the sink, as it increases the time every node has to wait for the next hop (parent) node before transmitting the data packet. The per-link latency will be uniformly distributed in $[0, T_w]$ with average $T_w/2$ (plus the time to transmit the packet). Coming back to the problems $(\boldsymbol{P1})$ and $(\boldsymbol{P2})$, the optimal solution of problem $(\boldsymbol{P1})$, $X_E^* = T_{w_E}^*$, will result in the pair $(E_{\text{best}}, L_{\text{worst}})$, where, $E_{\text{best}}=E(X_E^*)$ and $L_{\text{worst}}=L(X_E^*)$. On the other hand, the optimal solution of problem $(\boldsymbol{P2})$, $X_L^* = T_{w_L}^*$, will result in the pair $(E_{\text{worst}}, L_{\text{best}})$, where, $E_{\text{worst}}=E(X_L^*)$ and $L_{\text{best}}=L(X_L^*)$, Fig. 4.

Observing Fig. 4, there is a clear trade-off between minimizing energy consumption and latency in duty-cycled wireless MAC protocols. Deployments in which delay is not a requirement will use optimization problem $(\boldsymbol{P1})$ to model the MAC parameters at the cost of increasing delay. Deployments in which energy is not a requirement and low delay is a *must* will use optimization problem $(\boldsymbol{P2})$ to model the MAC parameters at the cost of increasing energy consumption. However, deployments in which both metrics are a *must* need to find a fair trade-off operational point.

In order to find the optimal trade-off solution for both optimization problems, we define a bargaining problem in which each optimization problem represents a player, i.e., player *Energy* and player *Latency*. The Bargaining problem is a problem of understanding how several agents should cooperate when non-cooperation leads to Pareto-inefficient results. The solution of the problem is one that coincides to the solution that an arbitrator would recommend. There exists various solutions[5] to the bargaining problem depending on the desired properties of the solution. Two of the most applied axiomatic solutions are the Nash Bargaining Solution (NBS) [Nash Jr 1950] and the Kalei-Smorodinsky Bargaining Solution (KSBS) [Kalai and Smorodinsky 1975].

3.5. The Nash Bargaining Solution (NBS) for Duty-Cycled MAC

A bargaining game with two players selects one of the possible player's outcomes of a joint collaboration [Nash Jr 1950] [Nissan et al. 2007]. Let A$\subset$R^2 be the set of alternatives the players face, let S$=\{s=(u_1(a),u_2(a)) \mid a\in A\}$ be the set of feasible utility payoffs, and let $v \in S$ be a disagreement or *threat point*. Each point in S corresponds to the outcome of the bargaining and specifies the utility for this outcome. The disagreement or threat point, $v=(v_1, v_2)$, represents the value that each player expects to receive if the negotiation breaks down. The goal of the bargaining is to choose a feasible agreement, $\Phi:(S, v) \to S$, that results from the negotiation.

The NBS considers that S is convex, compact, and there exists an $s\in S$ such that s$>$v for both players. Players have complete information over S, v. The NBS deals with the bargaining game by solving the following optimization problem when there are two players:

---

[5]There are other points that lie in the Pareto frontier and thus fulfil the E_{budget} and L_{max} constraints. However, these may not necessarily lead to an agreement by both players.
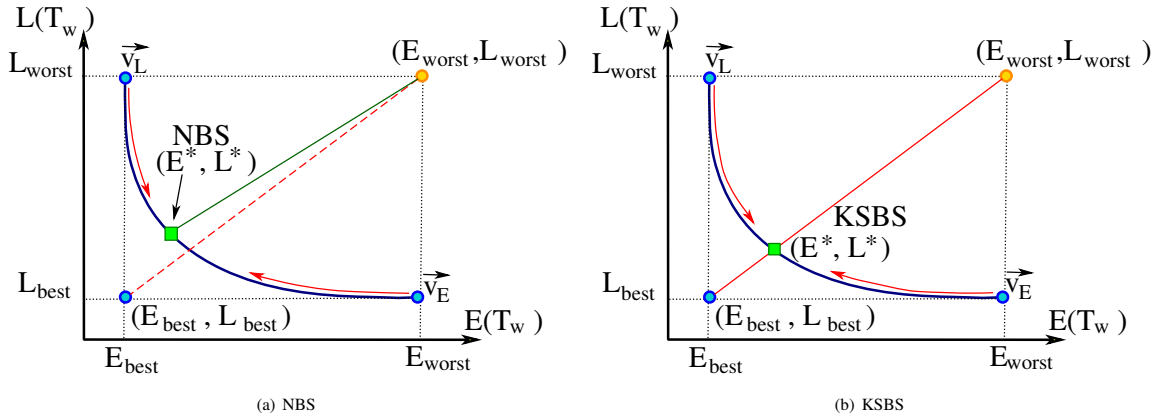
(a) NBS

(b) KSBS

Fig. 4. (a) Nash Bargaining Solution (NBS) and Kalei-Smorodinsky Bargaining Solution (KSBS) for Duty-Cycling MAC's.

$$\text{(NBS)} \quad \max \quad (s_1 - v_1)(s_2 - v_2)$$
$$\text{s.t.} \quad s \in S$$
$$(s_1, s_2) \geq (v_1, v_2)$$
$$\text{var.} \quad s.$$

The NBS has the following axioms, [Nash Jr 1950]: (i) *Pareto Optimality*, (ii) *Symmetry*, (iii) *Invariant to affine transformations*, and (iv) *Independence of Irrelevant Alternatives*. The Pareto efficiency axiom specifies that any bargaining solution, $\Phi(S, v)$, is preferred to a disagreement point. The symmetry axiom specifies that the utilities of the decision-makers at the solution are equal if the set S is symmetric. The invariance to affine transformations specifies that the solution is independent of the units of the decision-makers. Finally, the independence of irrelevant alternatives specifies that if the negotiation set S is narrowed to produce a smaller set, say S', then the solution $\Phi(S', v) = \Phi(S, v)$. The NBS specifies that there exists an optimal solution since S is compact and the objective function is continuous. The uniqueness of the optimal solution is guaranteed when the objective function is quasi-concave. The NBS, $\Phi(S, v)$, is the unique bargaining solution that satisfies the previous four axioms.

Let intervals $A_E = [E_{\text{worst}}, E_{\text{best}}]$ and $A_L = [L_{\text{worst}}, L_{\text{best}}]$ be the set of strategies[6] that respectively player energy and player delay may take, and $s_E \in A_E$, $s_L \in A_E$ the strategies chosen by the players. On the one hand, the threat strategy for player delay is to play strategy, L_{best}, that makes the player energy to get, E_{worst}. On the other hand, the threat strategy for player energy is to play strategy, E_{best}, that makes the player delay to get, L_{worst}. Let $v_E = E_{\text{worst}}$, and $v_L = L_{\text{worst}}$, be the threat values of each player, where each threat value represents the utility threshold of each player to sign the agreement. If no feasible solution is found, each player gets a cost $\Phi(S, v) = \infty$. Note that $E(X)$ and $L(X)$ are cost functions instead of utility functions, i.e., signs have to be reversed, and the term $(E(X), L(X)) \in S$ represents the intrinsic conditions that each MAC protocol has to fulfil. The (NBS) problem is expressed as:

$$\text{(P3-NBS)} \quad \max \quad (E_{\text{worst}} - E(X))(L_{\text{worst}} - L(X))$$
$$\text{s.t.} \quad (E_{\text{budget}}, L_{\text{max}}) \geq E(X), L(X)$$
$$(E_{\text{worst}}, L_{\text{worst}}) \geq E(X), L(X)$$
$$(E(X), L(X)) \in S$$
$$\text{var.} \quad X.$$

If a player unilaterally reduces its threat, and no feasible solution to the problem (P3-NBS) is found, each player obtains a cost of ∞. The (NBS) problem ensures that the solution belongs to the Pareto frontier (axiom (i)). This axiom states that a bargaining solution (s_1^*, s_2^*) is Pareto efficient if for any other solution $(s_1, s_2) \in S$, then $(s_1^*, s_2^*) \geq (s_1, s_2)$.

---

[6]The strategy is chosen by selecting the corresponding system parameters $X \in \Theta$. For example, player delay chooses $X = T_w$ that produces an utility of L_{best}. The player delay is said to choose the strategy L_{best}.

This condition, then, ensures that there is no feasible point $(s_1, s_2) \in$ S that is Pareto superior to the solution. A *Pareto Frontier* is the set of solutions that are Pareto efficient. Fig 4.a) shows how the NBS works. Each player can prevent the agreement threatening with a worst value or can reduce its threat looking for a feasible point that satisfies both players. The solution (E^*, L^*) of the optimization problem (**P3**) where $E^*=E(X^*)$ and $L^*=L(X^*)$ will be the optimal cost for both players under the agreement. However, although the solution of the Nash Bargaining Problem lies in the Pareto frontier, the axiomatic solution does not ensure the fairness of the solution. This means that in the obtained solution, one player may reduce/increase its cost/utility more than the other. As [Kalai and Smorodinsky 1975] illustrates, let us assume a normalized cooperative game with convex hull in the area $S_1=\{(1,0), (0,1), (3/4,3/4)\}$. The NBS solution with threat point (0,0) is (3/4,3/4). Let us now assume a convex hull in the area $S_2=\{(1,0), (0,1), (1,0.7)\}$. The NBS solution with threat point (0,0) is (1,0.7) but player 2 has reasons to demand more in $(0,S_2)$ than he does in $(0,S_1)$. A fair trade-off solution requires that each player reduces/increases its cost/utility function with the same percentage. The Kalai-Smorodinsky Bargaining Solution deals with this problem and defines an equity (fair) solution.

3.6. The Kalai-Smorodinsky Bargaining Solution (KSBS) for Duty-Cycled MAC

The main idea behind the KSBS, [Kalai and Smorodinsky 1975], is that each player maximizes its utility while obtaining the same fraction of utility as any other player. Let us define the ideal point [7] b(S,v) as the bargainers' expectations before coming to the negotiation table in the energy-delay game. The KSBS deals with the bargaining game by solving the following optimization problem when there are two players:

$$
\begin{aligned}
\text{(KSBS)} \quad \max \quad & r \\
\text{s.t.} \quad & s \in S \\
& (s_1, s_2) \geq (v_1, v_2) \\
& \frac{s_1 - v_1}{b_1 - v_1} = r \\
& \frac{s_2 - v_2}{b_2 - v_2} = r \\
\text{var.} \quad & s,
\end{aligned}
$$

The KSBS, [Kalai and Smorodinsky 1975], has the following axioms, (i) *Pareto Optimality*, (ii) *Symmetry*, (iii) *Invariant to affine transformations*, and (iv) *Monotonicity*. The KSBS modifies the NBS axioms by dropping the independence of irrelevant alternatives axiom and adding the monotonicity axiom. This axiom states, that if, for every utility level that player 1 may demand, the maximum feasible utility level that player 2 can simultaneously reach is increased, then the utility level assigned to player 2 according to the solution should also be increased. Naming the three first axioms as the standard axioms, [Nash Jr 1950] shows that there is a unique standard independent solution while [Kalai and Smorodinsky 1975] shows that there is a unique standard monotone solution, and both are incompatible. The Kalai-Smorodinsky theorem states that for a pair (S,v) the maximal element of S on the line v to b(S,v) is the solution of the KSBS.

Defining the ideal point as b(S,v)=$(E_{\text{best}}, L_{\text{best}})$, and remembering that E and L are cost functions, the KSBS game for the 2-players (energy and delay) is expressed as:

$$
\begin{aligned}
\text{(P3-KSBS)} \quad \max \quad & r \\
\text{s.t.} \quad & (E_{\text{budget}}, L_{\text{max}}) \geq E(X), L(X) \\
& (E_{\text{worst}}, L_{\text{worst}}) \geq E(X), L(X) \\
& \frac{E_{\text{worst}} - E(X)}{E_{\text{worst}} - E_{\text{best}}} = r \\
& \frac{L_{\text{worst}} - L(X)}{L_{\text{worst}} - L_{\text{best}}} = r \\
& (E(X), L(X)) \in S \\
\text{var.} \quad & X
\end{aligned}
$$

The KSBS solution ensures that the solution belongs to the Pareto frontier and that it lies in the segment that connects the worst point, $(E_{\text{worst}}, L_{\text{worst}})$, to the ideal point, $(E_{\text{best}}, L_{\text{best}})$, (Fig 4.b), which provides an equity solution:

---

[7]The ideal point is also called the aspiration point or utopia point by many authors, e,g [Balakrishnan et al. 2011].

$$\frac{E_{\text{worst}} - E^*}{E_{\text{worst}} - E_{\text{best}}} = \frac{L_{\text{worst}} - L^*}{L_{\text{worst}} - L_{\text{best}}} \qquad (3)$$

4. APPLICATION TO A SET OF WSN MAC PROTOCOLS

We apply the optimization framework to four state-of-the-art energy-delay efficient MAC protocols, B-MAC [Polastre et al. 2004], X-MAC [Buettner et al. 2006], RI-MAC [Sun et al. 2008], SMAC [Ye et al. 2004], DMAC [Lu et al. 2007], and LMAC [van Hoesel and Havinga 2004] as representatives of the main categories of duty-cycled MAC protocols, preamble sampling, beacon-based, slotted contention-based, and frame-based respectively. These protocols are considered as canonical MAC in [Langendoen and Meier 2010], and we refer to the elaborated survey of energy-efficient MAC protocols available online [Sou 2011], where most of the recent protocols extend upon their canonicals like ContikiMAC [Dunkels 2011], A-MAC [Dutta et al. 2012], and CyMAC [Peng et al. 2011] which make the analysis valid for them. The choice of these protocols is to exemplify the framework and show its usefulness to optimize different MAC parameters that permit to achieve a fair energy-delay trade-off. For the sake of clarity, the formulation derived by the analysis made in [Langendoen and Meier 2010] is used. Some of the formulas were obtained by the original authors of the proposed protocols, while others are derived in that paper. For the sake of space reduction, we summarize the formulation of only the RI-MAC protocol, while those of B-MAC, X-MAC, SMAC, DMAC, and LMAC are given in Appendix A. The results of optimization and simulations are provided for the six MAC protocols.

4.1. Protocol Description

RI-MAC (Beacon-based) [Sun et al. 2008] is a receiver-initiated asynchronous protocol that tries to reduce the amount of time a pair of nodes occupy the medium by preambles before they reach a rendezvous time for data exchange. This is to reduce the global network delivery delay. Every node periodically wakes-up and broadcasts a beacon after T_w (1), Fig. 5. When a node wants to send a data packet, it stays silently active for a period of T_{wait} (2) until the wake-up of its receiver, and it starts contending to send its packets upon receiving a beacon from its receiver (3). Multiple nodes can contend in the contention windows T_{cw}, where the receiver keeps its radio on waiting for eventual data packets and returns back to sleep after T_{timeout}. The winer among the contending nodes transmits the data packet, T_{data}, which spans for the transmission of the header and the payload (4). The receiver acknowledges the data packet with another beacon which spans for the transmission of the wake-up beacon $T_{\text{ack}} = T_{\text{beacon}}$ (5). Note that this beacon's role is twofold: first, it acknowledges the correct receipt of the sent data packet, and second, it invites for new data packets from other senders before returning back to sleep. Following the defined energy model in section 3.3, the energy consumption is modeled by the effective duty cycle (i.e., the fraction of time the radio is switched on). Knowing the duty-cycle, the node's lifetime can be easily determined given the current draws, I_{on}, I_{off}, of the radio respectively in active and sleep modes. For example, let E^n be the effective duty-cycle (energy consumption), and Q be the battery capacity of a node n, then the node's lifetime can be calculated by $T^n = Q/(E^n I_{\text{on}} + (1 - E^n) I_{\text{off}})$ [Zimmerling et al. 2012]. This allows to omit the physical energy consumption expression and only focus on the effective duty-cycle aspects [Langendoen and Meier 2010]. Following the RI-MAC operation model, the key adjustable parameter that affects the energy and delay performance is the wake-up period, T_w. The vector parameter for RI-MAC protocol is thus given by $X_{\text{RI-MAC}} = [T_w]$. The per-node energy consumption based on the protocol operation modes, the e2e packet delay, and the bottleneck constraint[8] are provided in the following equations (the description of every term used in the formulas can be found in table I and table II).

a) The Energy of node n at level d:

Remember that the average input traffic, F_I^n, output traffic, F_{out}^n, and background traffic, F_B^n, for a node n at level d, are respectively defined as, $F_I^n = F_I^d|_{n \in d}$, $F_{out}^n = F_{out}^d|_{n \in d}$, and $F_B^n = F_B^d|_{n \in d}$. In RI-MAC, the energy is consumed each time a node performs beacon sending, E_{beacon}, data transmission, E_{tx}, data reception, E_{rx}, and when in traffic overhearing mode, E_{ovr}[9]:

$$E^n = E_{\text{beacon}}^n + E_{\text{tx}}^n + E_{\text{rx}}^n + E_{\text{ovr}}^n, \qquad (4)$$

where $\quad E_{\text{beacon}}^n = \frac{T_{\text{up}} + T_{\text{beacon}}}{T_w}, \quad T_{\text{wait}} = T_w/2, \quad T_{\text{tx}} = T_{\text{cs}} + T_{\text{data}} + T_{\text{ack}}, \quad E_{\text{tx}}^n = (T_{\text{wait}} + T_{\text{tx}}) F_{\text{out}}^n, \quad E_{\text{rx}}^n = (T_{\text{timeout}}/2 + T_{\text{cs}} + T_{\text{data}} + T_{\text{ack}}) F_I^n,$ and $E_{\text{ovr}}^n = \frac{T_{\text{tx}}}{T_w} (T_{\text{cs}} + T_{\text{hdr}}) F_B^n$.

---

[8]Since the defined traffic model is targeted to low data rate applications, this constraint is used to avoid the bottleneck at the sink.

[9]The duty-cycle is calculated by multiplying the time spent in a given mode by the frequency of its occurrence (e.g. $1/T_w$, F_{out}^n, etc.)
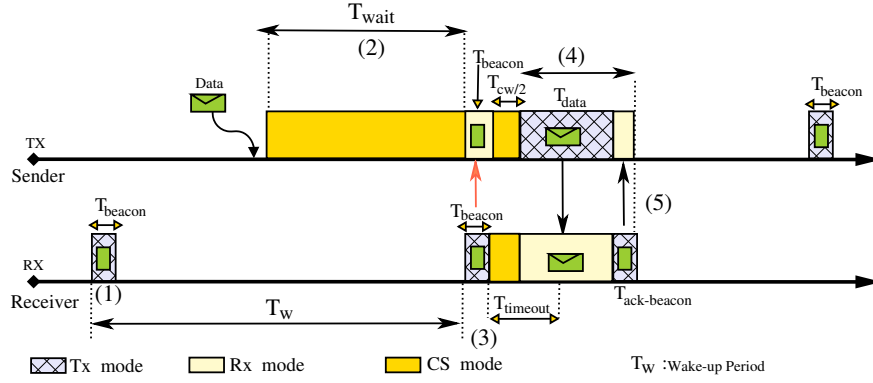
Fig. 5.   RI-MAC's beaconing, waiting in idle listening, transmitting, and receiving modes.

The energy consumption, E^n, is a per-node function that depends on the intrinsic parameters of the MAC protocol, the traffic it generates and it relays. Therefore, since nodes belonging to the same ring d will generate and relay, on average, the same traffic, they will consume the same energy.

b) The delay of node n at level d:
The average e2e delay of a node is determined by the summation of one-hop delays, which include the waiting time before the receiver wakes-up, the time to receive the beacon, the contention time, the time to send and knowledge the data packet.

$$L^n = \sum_{i=1}^{d} \left(\frac{T_w}{2} + T_{\text{beacon}} + \frac{T_{cw}}{2} + T_{\text{data}} + T_{\text{ack}} \right), \tag{5}$$

where $T_{\text{data}} = T_{\text{hdr}} + P/R$. Again, all nodes at the same ring d will have the same average e2e delay.

c) The bottleneck constraint:

$$|I^0| \, E_{tx}^1 < 1/4, \tag{6}$$

where I^0 is the number of input links at level 0 (at the sink). From equations (4-6), the following system wide energy-delay functions are defined where the non-tunable parameters are grouped as constants:

a) The network energy consumption function:

$$E^{\text{RI-MAC}} = \max_{n \in N}(E^n) = \max_{n \in N}(\frac{\eta_1}{T_w} + \eta_2 T_w + \eta_3) \tag{7}$$

where $\eta_1 = T_{\text{up}} + T_{\text{beacon}} + (T_{cs} + T_{\text{data}} + T_{\text{ack}})(T_{cs} + T_{\text{hdr}})F_B^n$, $\eta_2 = F_{\text{out}}^n/2$, and $\eta_3 = (T_{cs} + T_{\text{data}} + T_{\text{ack}})F_{\text{out}}^n + (T_{\text{timeout}}/2 + T_{cs} + T_{\text{data}} + T_{\text{ack}})F_I^n$.

b) The e2e packet delay function:

$$L^{\text{RI-MAC}} = \max_{n \in N}(L^n) = \max_{n \in N}(\varepsilon_1 T_w + \varepsilon_2) \tag{8}$$

where $\varepsilon_1 = \sum_{i=1}^{d} 1/2$ and $\varepsilon_2 = \sum_{i=1}^{d}(T_{\text{beacon}} + \frac{T_{cw}}{2} + T_{\text{data}} + T_{\text{ack}})$.

Table II. RI-MAC Symbols used in Energy & Delay Equations.

MAC	Parameter & Description	Values
RI-MAC	T_w: RI-MAC wake-up period [ms] T_{beacon}: Wake-up beacon duration [ms] T_{wait}: Waiting time until the receiver wake-up [ms] T_{cw}: Contention window size [ms] T_{timeout}: Waiting time for incoming data packets [ms] $T_{\text{hdr}}, T_{\text{ack}}$: Pkt header & Ack duration [ms]	T_w^* $\frac{9+L_{\text{pbl}}}{R}$ $T_w/2$ $15 * 0.62$ T_{cw} $\frac{9+L_{\text{pbl}}}{R}$

Let us define the network lifetime as the time until the first battery exhaustion of any node. Nodes that are placed in the nearest ring to the sink (d=1) are the ones that convey more traffic towards the sink, since they have to forward their own traffic and the traffic that flows from all the other rings ($d > 1$). These nodes are the most energy consuming and they will be the first ones to die. Since all nodes at $d = 1$ have the same traffic on average, Eq(7) reduces to:

$$E^{\text{RI-MAC}} = \max_{n \in N}(E^n) = \frac{\eta_1}{T_w} + \eta_2 T_w + \eta_3 \tag{9}$$

with η_1, η_2 and η_3 having $F_B^n = F_B^d|_{d=1}$, $F_{out}^n = F_{out}^d|_{d=1}$ and $F_I^n = F_I^d|_{d=1}$.

On the other hand, the maximum e2e delay occurs in nodes placed at the outer ring d=D. Then, Eq(8) reduces to[10]:

$$L^{\text{RI-MAC}} = \max_{n \in N}(L^n) = \varepsilon_1 T_w + \varepsilon_2 \tag{10}$$

with ε_1=$D/2$ and ε_2=$D(T_{\text{beacon}} + \frac{T_{\text{cw}}}{2} + T_{\text{data}} + T_{\text{ack}})$.

4.2. Framework Application

Before applying the game theory framework, the energy consumption and the e2e delay resulted for different values of the wake-up period T_w, the slot size T_{slot} (T_{slot} =$T_{\text{sync}} + T_{\text{active}} + T_{\text{sleep}}$), and frame size T_{frame} of B-MAC, X-MAC, RI-MAC, SMAC, DMAC, and LMAC, respectively, are plotted in Fig. 6. It can be observed that each protocol reduces its average e2e delay at the cost of rising energy consumption, and vice versa, as predicted in Fig 4. For analyzing this trade-off, the general optimization problems P1 and P2 are first used to determine the best parameters' values that permit to achieve optimal energy and delay objectives. In the following, RI-MAC is used as a baseline example for the framework application.

The network designer, with the knowledge of the network topology and the sampling rate, may characterize functions for the energy consumption, $E^{\text{RI-MAC}}$, and e2e delay, $L^{\text{RI-MAC}}$, as described in Eq(9) and Eq(10). The objective of the network designer is to obtain the optimal value for the tuning parameter, T_w, that minimizes energy consumption and e2e delay in a fair manner. For that purpose, the specific energy consumption optimization problem ($P4$) is firstly defined to obtain the threat value $v_L^{\text{RI-MAC}} = L_{worst}^{\text{RI-MAC}}$. Secondly, the specific e2e delay optimization problem ($P4'$) is defined to obtain the threat value $v_E^{\text{RI-MAC}} = E_{worst}^{\text{RI-MAC}}$. The NBS optimization problem ($P4^*$) is then defined and solved. Finally, the KSBS is solved with a heuristic from the NBS optimization problem. The optimal value, T_w^*, used by the network designer is the value given by the optimization problem ($P4^*$) if the NBS axioms are met, or the value given by the heuristic if the KSBS axioms are met.

*4.2.1. **Energy Optimization**.* Given the application requirements in terms of e2e packet delay bound $L_{\max}$, energy optimization derives optimal MAC parameters that give the minimal network energy consumption subject to $L_{\max}$:

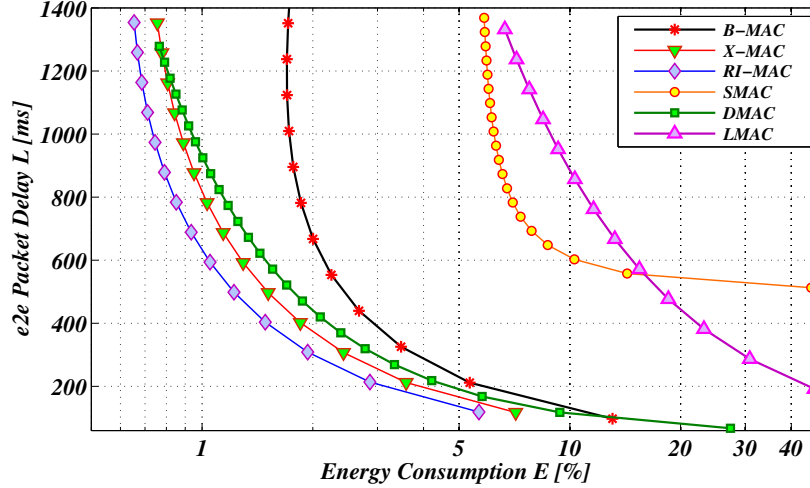[10]The same hypothesis is used in the protocols described in the appendix.

Fig. 6. e2e Delay vs. Energy Consumption for different values of the wake-up period T_w, slot size T_{slot}, and frame size T_{frame} of B-MAC, X-MAC, RI-MAC, SMAC, DMAC, and LMAC.

$$
\begin{aligned}
\text{(P4)} \quad &\min \quad E^{\text{RI-MAC}}(T_w) \\
&\text{s.t.} \quad L^{\text{RI-MAC}}(T_w) \leqslant L_{\max} \\
&\qquad\;\; T_w \geqslant T_w^{\min} \\
&\qquad\;\; |I^0|\, E_{tx}^1 \leqslant 1/4 \\
&\text{var.} \quad T_w
\end{aligned}
$$

Eq(9) and Eq(10) are posynomials, while the second and third constraint of $(\boldsymbol{P4})$ are monomials. A monomial is an expression of the type $c\,x_1^{a^{(1)}} x_2^{a^{(2)}} ... x_n^{a^{(n)}}$, with $c \geq 0$ and $a^{(j)} \in R$ for j=1,...,n, while posynomials are summation of monomials. It is to say, an expression of the type $\sum_{k=1}^{K} c_k x_1^{a_k^{(1)}} x_2^{a_k^{(2)}} ... x_n^{a_k^{(n)}}$, with $c_k \geq 0$ for $k = 1, ..., K$ and $a^{(j)} \in R$ for $j = 1, ..., n$. Optimization problem $(\boldsymbol{P4})$ is a geometric optimization problem that is non-convex and non-linear. A geometric problem is a problem in which the objective function and the inequality constraints are posynomials and the equality constraints are monomials. Geometric problems may be easily transformed to convex problems, [Boyd and Vandenberghe 2004], using a logarithmic change of variables and a logarithmic transformation of the objective and constraint functions. Let us define vector $a_k = (a_k^{(1)}, ..., a_k^{(n)})$, vector x=$(x_1,...,x_n)$, y_j=$\log(x_j)$, and b_k=$\log(c_k)$. Posynomial $\sum_{k=1}^{K} c_k x_1^{a_k^{(1)}} x_2^{a_k^{(2)}} ... x_n^{a_k^{(n)}}$ is transformed in a log-sum-exp function of the type $\log \sum_{k=1}^{K} e^{(b_k + a_k^T y)}$, where vector a_k^T is the transpose of vector a_k. Monomials are transformed in linear expressions of the type b+a^Ty. It may be easily proved that log-sum-exp functions are convex, and then, optimization problem $(\boldsymbol{P4})$ may be transformed in a convex form. Solving now problem $(\boldsymbol{P4})$, the optimal point $X^*_{\text{RI-MAC}}$ =$[T_w^*]$ is obtained, the optimal value of $(\boldsymbol{P4})$ is $E^{\text{RI-MAC}}_{\text{best}}$= $E^{\text{RI-MAC}}(X^*_{\text{RI-MAC}})$, and the corresponding e2e packet delay is obviously non-optimal, $L^{\text{RI-MAC}}_{\text{worst}}$= $L^{\text{RI-MAC}}(X^*_{\text{RI-MAC}})$. Energy optimization for B-MAC, X-MAC, SMAC, DMAC and LMAC protocols is provided in the appendix (Sec.A.2.1) by solving problems $(\boldsymbol{P5})$, $(\boldsymbol{P6})$, $(\boldsymbol{P7})$, $(\boldsymbol{P8})$, and $(\boldsymbol{P9})$ respectively. Problems $(\boldsymbol{P5})$-$(\boldsymbol{P8})$ again are geometric optimization problems. On the other hand, the energy consumption for LMAC is in the form of sum of linear fractions and optimization problem $(\boldsymbol{P9})$ is not a geometric optimization problem. However, this problem may be transformed to an equivalent convex problem using a primal-relaxed dual global optimization approach, [Floudas and Visweswaran 1993] [Benson 2004].

[Floudas and Visweswaran 1993] propose a deterministic global optimization approach for non-convex constrained non-linear programming problems. The idea is to decompose the original non-convex problem into primal and relaxed dual subproblems by introducing new transformation variables if necessary and partitioning of the resulting variable

set. Partitioning of the variable set means to divide the variable set into two parts in such a way that the objective and constraint functions are convex for one part of the variables given that the other part is fixed. Then, the projection of the problem into the space of a subset of the variables results in a convex programming problem. The method is useful in quadratic optimization problems with linear and/or quadratic constraints, and in optimization problems involving polynomial functions of one or more variables in the objective function and/or the constraint set. Since the primal problem still may be non-convex in the projected set, [Floudas and Visweswaran 1993] propose the use of the dual representation and the relaxation of the constraint region for the problem, thus obtaining a lower bound for the solution of the problem. Finally, iterating the process, a global solution is determined. [Benson 2004] applies the [Floudas and Visweswaran 1993] process to a non-convex optimization problem with sum of linear fractions. Let us consider the following optimization problem:

$$(\text{PN}) \quad \min \ \sum_{k=1}^{K} \frac{(a_k^T x + b_k)}{(c_k^T x + d_k)}$$
$$\text{s.t.} \quad x \in X,$$

where $X \subseteq R^n$ is a non-empty, compact convex set, and $a_k^T x + b_k$, $c_k^T x + d_k$ are positively-valued functions on X. In order to solve this problem, [Benson 2004], let us define:

$$
\begin{aligned}
m_k &= \min_{x \in X} (c_k^T x + d_k) \\
M_k &= \max_{x \in X} (c_k^T x + d_k) \\
L_k &= 1.0/M_k \\
LL_k &= 1.0/m_k
\end{aligned}
$$

Then, x^* is a globally optimal solution for problem (PN) if and only if, for some $y^* \in R^K$, (x^*, y^*) is a global optimal solution for the problem:

$$(\text{PN1}) \quad \min \ \sum_{k=1}^{K} (a_k^T x + b_k)$$
$$
\begin{aligned}
\text{s.t.} \quad & -y_k(c_k^T x + d_k) + 1.0 \leq 0 && k=1,..,K \\
& -y_k - LL_k \leq 0 && k=1,..,K \\
& -y_k + L_k \leq 0 && k=1,..,K \\
& x \in X
\end{aligned}
$$

Finally, [Floudas and Visweswaran 1993] gives an algorithm that solves via branch and bound optimization problems such as (PN1). Fig 7.a shows the optimal solution obtained for each protocol when varying the delay bound $L_{\max}$ from $500ms$ to $3000ms$ using the network topology model and traffic model described in section 3.1. Optimization problems $(P4)$-$(P8)$ have been solved using CVX, a Matlab-based modeling system for disciplined convex optimization that supports geometric programming (GP), [Boyd and Vandenberghe 2004]. Optimization problem $(P9)$ has been solved using the [Benson 2004] primal-relaxed dual global optimization approach. It is observed that by relaxing the $L_{\max}$ constraint, the energy consumption is lowered for all MAC protocols until a given value of $L_{\max}$ where there is no space for improvement. The difference between protocols is due to the intrinsic design of the MAC protocols and it is out of the scope of this paper.

Fig. 7.{b-e} show how the optimal point values T_w^* (B-MAC, XMAC and RI-MAC), T_{active}^* and T_{sleep}^* (SMAC), T_{sync}^* (DMAC), and T_{frame}^* (DMAC and LMAC), increase as the e2e packet delay bound $L_{\max}$ increases. From the figures, it can be observed that the increase at the beginning was proportional to the e2e delay bound (low values of $L_{\max}$), but it becomes stable for some protocols. This means that relaxing the e2e delay bound beyond those values have no effect on the energy optimization parameters, as well as the consumed energy. The reason again is due to the intrinsic behavior of the MAC protocols. For example, taking the RI-MAC protocol, relaxing the $L_{\max}$ bound would allow to increase the T_w^* and thus the node would stay longer in the sleep mode, saving energy. However, a transmitting node will have to wait in the idle state for more time to transmit a packet, increasing the energy consumption. Similar behavior occurs at the other MAC protocols.

*4.2.2. **Delay Optimization**.* Now, given the application requirements in terms of maximum energy budget E_{budget} expressed as the maximum allowed duty cycle, we are interested in finding the optimal MAC parameters that give the

Fig. 7.   (a) The optimized Energy E_{best} for all the MACs, (b) the wake-up period T_w for B-MAC, X-MAC, and RI-MAC, and (c) the active period T_{active} and sleep period T_{sleep} for SMAC, and (d) the synchronization period T_{sync} for DMAC and (e) the frame size T_{frame} for DMAC and LMAC obtained by varying $L_{\text{max}} \in [500, 3000]$ ms.

minimum e2e packet delay subject to maximum energy budget:

$$\text{(P4')} \quad \min \quad L^{\text{RI-MAC}}(T_w)$$
$$\text{s.t.} \quad E^{\text{RI-MAC}}(T_w) \leqslant E_{\text{budget}}$$
$$T_w \geqslant T_w^{\min}$$
$$|I^0| \, E_{tx}^1 \leqslant 1/4$$
$$\text{var.} \quad T_w$$

The delay optimization problems $(\boldsymbol{P4'})$-$(\boldsymbol{P9'})$ can be solved similarly to the energy optimization. Let $Y_{\text{RI-MAC}}^*$ $=[T_w^*]$ denotes the optimal point of problem $(\boldsymbol{P4'})$. The optimal delay value of problem $(\boldsymbol{P4'})$ is denoted by $L_{\text{best}}^{\text{RI-MAC}}$ $=L^{\text{RI-MAC}}(Y_{\text{RI-MAC}}^*)$. The corresponding energy consumption is obviously non-optimal, $E_{\text{worst}}^{\text{RI-MAC}}= E^{\text{RI-MAC}}(Y_{\text{RI-MAC}}^*)$.

Fig. 8.a shows the optimal solution obtained for each protocol when varying the maximum energy budget E_{budget} from 1.2% to 20%. The e2e packet delay is lowered for all MAC protocols as the maximum energy budget increases up to a given value, from which there is no more delay reduction for some protocols such as B-MAC, X-MAC, and RI-MAC. The difference in e2e packet delay between both protocols is again due to the intrinsic design of the MAC protocols, and it is out of the scope of this paper. Fig. 8.{b-e} depict the decrease of the optimal point values of T_w^* (B-MAC, XMAC and RI-MAC), T_{active}^* and T_{sleep}^* (SMAC), T_{sync}^* (DMAC), T_{frame}^* (DMAC and LMAC) as a function of the energy budget E_{budget}. All the figures show that the decrease is significant at the beginning (low values of E_{budget}), but it becomes insignificant as E_{budget} increases. This means that increasing the maximum energy budget beyond those values have no effect on the delay optimization parameters, as well as the achieved delay. The main reason of this behavior, taking again RI-MAC as example, is the $T_w \geq T_w^{\min}$ constraint, where T_w cannot be decreased as much as E_{budget} is increased, since the T_w constraint has to be fulfilled. Similar constraints hold for the other MAC protocols.
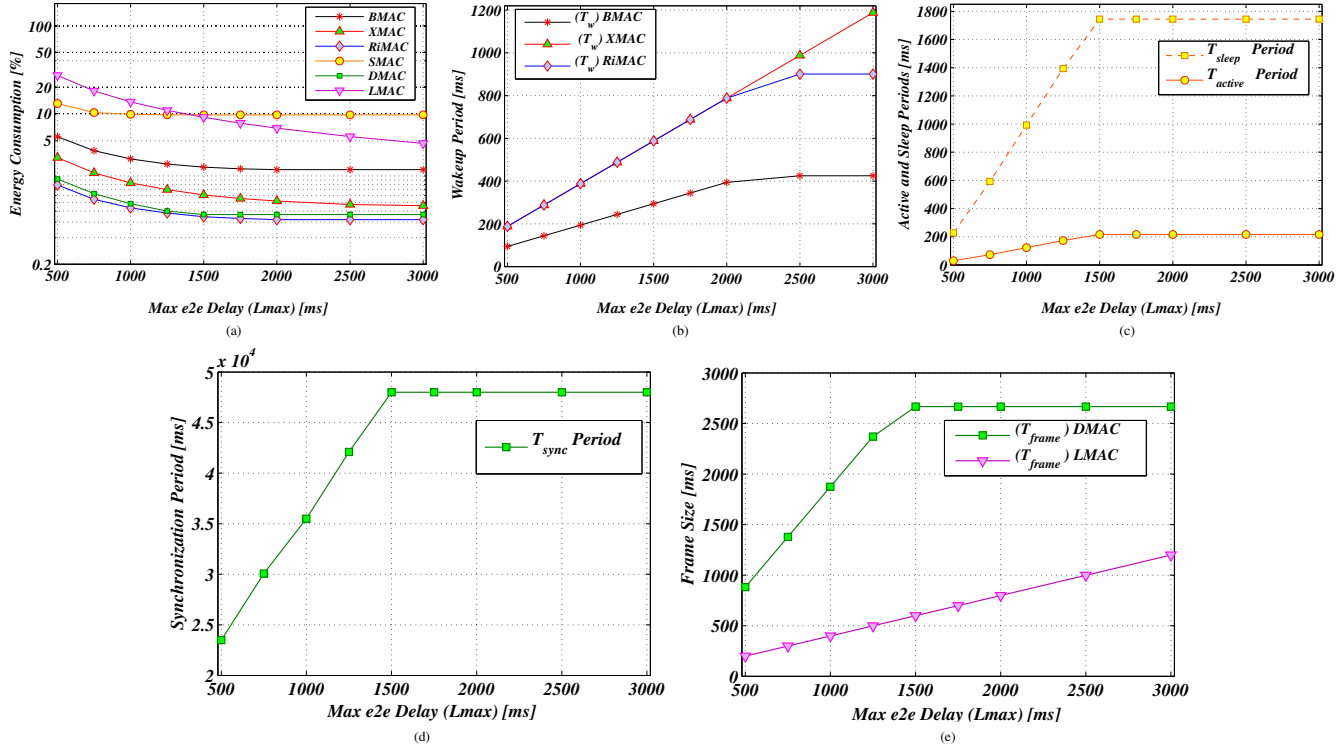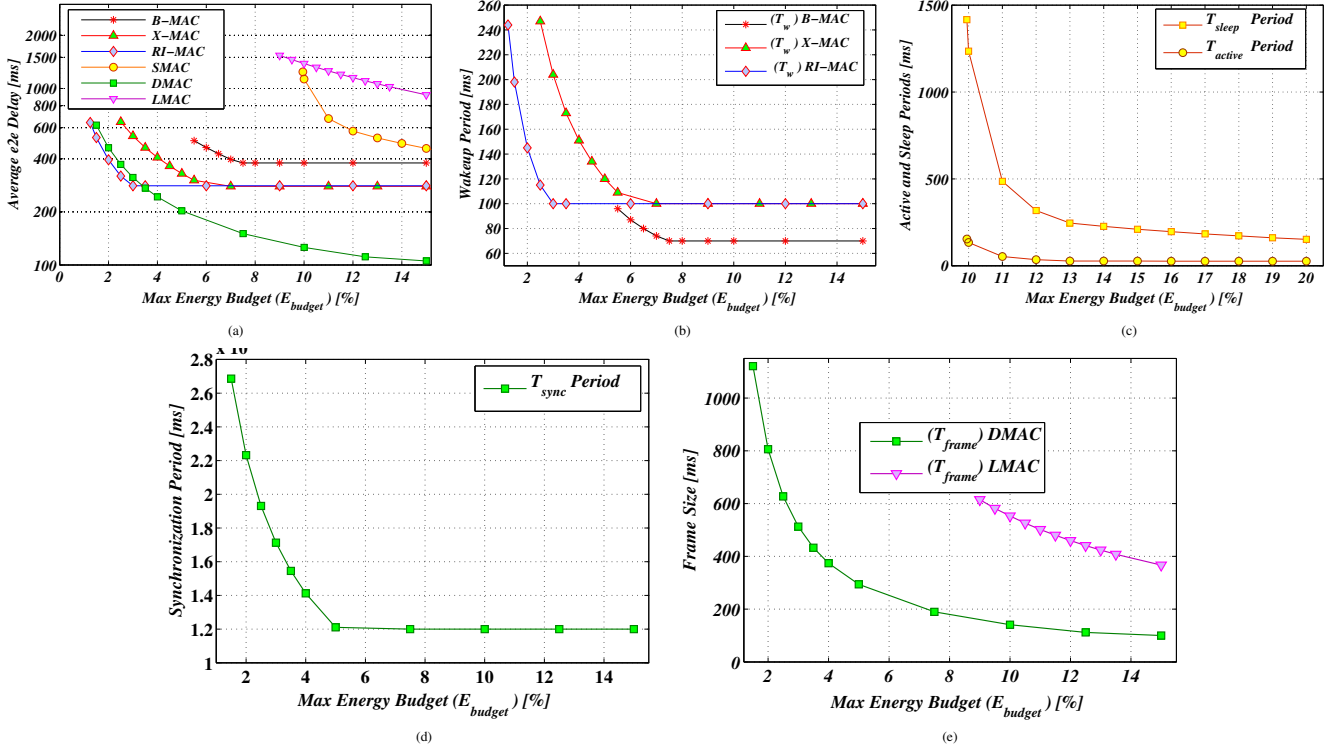
Fig. 8. (a) The optimized e2e Delay L_{best} for all the MACs, (b) the wake-up period T_w for B-MAC, X-MAC, and RI-MAC, and (c) the active period T_{active} and sleep period T_{sleep} for SMAC, and (d) the synchronization period T_{sync} for DMAC and (e) the frame size T_{frame} for DMAC and LMAC obtained by varying $E_{\text{budget}} \in [1.25, 20]$ %.

4.2.3. ***Energy-Delay Trade-off: NBS model***. The Nash Bargaining solution (P3-NBS) is applied in order to find an energy-delay trade-off. Let the point $(E_{\text{worst}}^i, L_{\text{worst}}^i)$ be the disagreement point, with $i=\{$BMAC, XMAC, RI-MAC, SMAC, DMAC, LMAC$\}$. The problem (P3-NBS) is non-linear non-convex, but this kind of problems can be transformed into a standard convex optimization problem without changing its solution [Zhao et al. 2013]. The idea is to define auxiliary variables E_1 and L_1 such that $E_1 \geq E^i(X)$ and $L_1 \geq L^i(X)$, which should be satisfied by the optimal solution. The proof comes from the fact that in order to attain the maximum in the objective function, E_1 and L_1 have to be minimum, and thus, be equal to $E^i(X)$ and $L^i(X)$ respectively (second constraint). Moreover, since $E^i(X)$ and $L^i(X)$ are less or equal than $(E_{\text{worst}}^i, L_{\text{worst}}^i)$, (first constraint), the solution is feasible whenever the problem (P3-NBS) is feasible, and application of (P3-NBS) to the MAC protocols yields a concave problem.

$$
\begin{aligned}
\text{(P3*-NBS)} \quad \max \quad & log(E_{\text{worst}}^i - E_1) + log(L_{\text{worst}}^i - L_1) \\
\text{s.\,t.} \quad & (E_{\text{worst}}^i, L_{\text{worst}}^i) \geqslant (E^i(X), L^i(X)) \\
& (E_1, L_1) \geqslant (E^i(X), L^i(X)) \\
& (E_1, L_1) \leq (E_{\text{budget}}, L_{\text{max}}) \\
& (E_1, L_1) \in S \\
\text{var.} \quad & E_1, L_1, X
\end{aligned}
$$
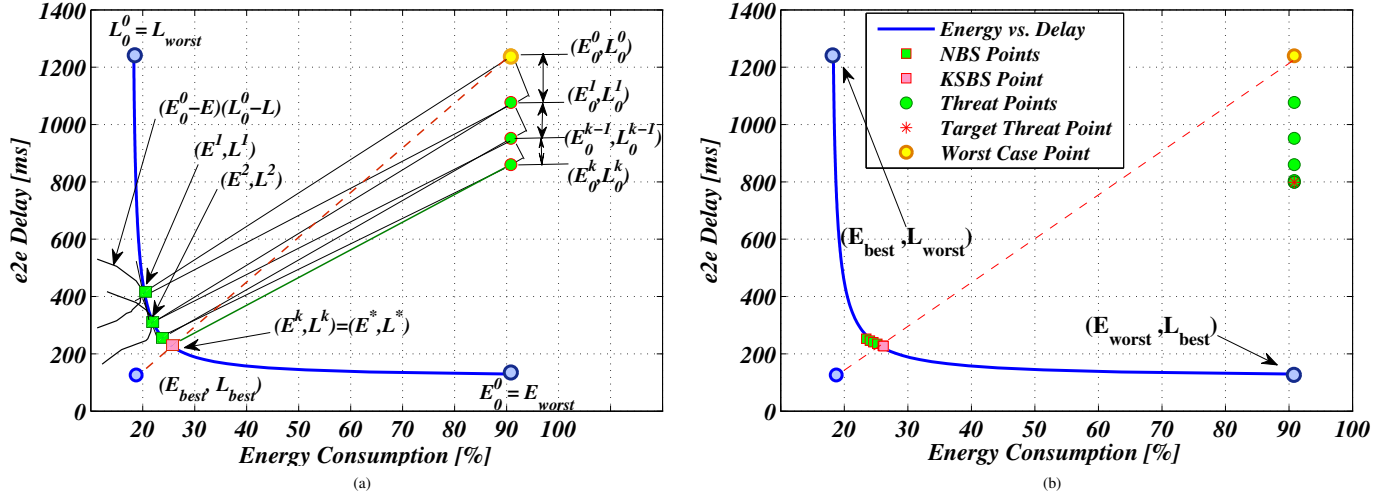
Fig. 9. (a) KSBS solution obtained from iterating over the NBS problem. (b) execution trace of the KSBS algorithm showing different threat and trade-off points before achieving the target threat point and the KSBS optimal point for SMAC protocol.

Consequently, the equivalent concave problem for RI-MAC is:

$$
\begin{aligned}
\text{(P4*)} \quad \max \quad & log(E_{\text{worst}}^{\text{RI-MAC}} - E_1) + log(L_{\text{worst}}^{\text{RI-MAC}} - L_1) \\
\text{s.t.} \quad & E_{\text{worst}}^{\text{RI-MAC}} \geqslant E^{\text{RI-MAC}}(T_w) \\
& E_1 \geqslant E^{\text{RI-MAC}}(T_w) \\
& L_{\text{worst}}^{\text{RI-MAC}} \geqslant L^{\text{RI-MAC}}(T_w) \\
& L_1 \geqslant L^{\text{RI-MAC}}(T_w) \\
& T_w \geqslant T_w^{\text{min}} \\
& |I^0| E_{tx}^1 \leqslant 1/4 \\
\text{var.} \quad & E_1, L_1, T_w
\end{aligned}
$$

*4.2.4. **Energy-Delay Trade-off: KSBS model**.* The problem (P3-KSBS) is non-linear and non convex. Due to the difficulty in finding an equivalent convex problem like in the NBS optimization problem, we provide an iterative method using the NBS model that converges to the KSBS solution. Let (E_0^0, L_0^0) be the initial threat value of each player [11]. Whenever a player has larger gain than the other, this will be due to the fact that its threat point is better than its adversary's threat point. Thus, the player with lower gain has to decrease its threat value [12], while the player with larger gain maintains its threat value. Let (E^k, L^k) be the optimal point obtained by the NBS model at the k-th step. The objective is to use a new initial threat value (E_0^k, L_0^k) such that the new NBS optimal point approximates the KSBS optimal point. Fig. 9(a) illustrates how the solution is obtained. Let the error δ^k be defined as the difference between the gains obtained by each player at the k-th step using the NBS model:

$$
\delta^k = \left| \frac{E_{worst} - E^k}{E_{worst} - E_{best}} - \frac{L_{worst} - L^k}{L_{worst} - L_{best}} \right| \tag{11}
$$

Now, the threat value of the player that results in less gain is decreased by a factor corresponding to the absolute difference between the gains obtained by each player at that iteration. This is illustrated in Algorithm 1 and Fig. 9.(a). In

---

[11]The point $(E_{\text{worst}}, L_{\text{worst}})$ is a first candidate as initial threat point (E_0^0, L_0^0). However, since the Pareto frontier is convex, the middle point $(\frac{E_{\text{best}} + E_{\text{worst}}}{2}, \frac{L_{\text{best}} + L_{\text{worst}}}{2})$ also lies in the line connecting $(E_{\text{worst}}, L_{\text{worst}})$ and $(E_{\text{best}}, L_{\text{best}})$, and it is nearer to the optimal KSBS point. Thus, selecting this point as initial threat point will speed up the convergence of the algorithm.

[12]Decrease the threat value because L and E are cost functions.

Table III. Convergence of the KSBS mechanism in SMAC. Initial threat point (E^0, L^0)=$(E_{\text{worst}}, L_{\text{worst}})$.

	k	0	1	2	3	4	5	6	7
	E_0^k	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000
	L_0^k	1239.9	1077.4	952.1	860.9	801.3	795.7	795.4	795.4
KSBS algorithm	E^k	0.2444	0.2498	0.2554	0.2611	0.2708	0.2719	0.2719	0.2719
	L^k	287.91	280.22	273.79	268.24	259.53	258.64	258.593	258.591
	δ^k	0.0655	0.0505	0.0368	0.0239	0.0024	0.0001	0.0000	0.0000

the first step, the NBS optimal point is obtained from the initial threat point (E_0^0, L_0^0)=$(E_{\text{worst}}, L_{\text{worst}})$ as the supporting hyperplane at (E^0, L^0) of the objective function $(E_0^0$-E$)(L_0^0$-L$)$ with the Pareto frontier. Since the gain of the delay player is less than the gain of the energy player, the threat point L_0^0 is decreased by a factor of $2\delta^0$, where δ^0 is the difference between the two obtained gains. In the following steps, the threat point of the player that obtains less gain, in our example the delay player, is decreased proportionally to δ^k. The new NBS optimal points with the new threat points converge to the KSBS optimal point. Fig. 9.(b) and Table III shows how the proposed algorithm converges fast after 8 iterations for SMAC protocol with an error lower than 10^{-5}, where L_{max}=5000ms, and $E_{budget} = 90\%$. Similar convergence results have been obtained for the other MAC protocols.   Fig. 10.{a-f} plot the results for the

---

ALGORITHM 1: KSBS optimal point.

Input: Problem (P3*-NBS), E_{worst}, L_{worst}, E_{best}, and L_{best}.
Output: The KSBS solution (E^*, L^*)=(E^k, L^k) and the optimal corresponding parameters X^*.
Set ϵ as a stop criterion value;
$k = 0$;
Set the initial threat point (E_0^0, L_0^0);
repeat

    Solve (P3*-NBS) max $log(E_0^k - E_1) + log(L_0^k - L_1)$

                s.t. $(E_0^k, L_0^k) \geqslant (E(X), L(X))$

                      $(E_1, L_1) \geqslant (E(X), L(X))$

                      $(E_1, L_1) \leq (E_{\text{budget}}, L_{\text{max}})$

                      $(E_1, L_1) \in S$

             var. E_1, L_1, X

  $E^k = E_1$;
  $L^k = L_1$;
  $k = k + 1$;
  $\delta^k = \big| \frac{E_{\text{worst}} - E^k}{E_{\text{worst}} - E_{\text{best}}} - \frac{L_{\text{worst}} - L^k}{L_{\text{worst}} - L_{\text{best}}} \big|$;
  if $\big(\frac{E_{worst} - E^k}{E_{worst} - E_{best}} < \frac{L_{worst} - L^k}{L_{worst} - L_{best}} \big)$ **then**

    $E_0^k = E_0^{k-1}$ - $2E_0^0\delta^k$;
    $L_0^k = L_0^{k-1}$;
  else
    $L_0^k = L_0^{k-1}$ - $2L_0^0\delta^k$;
    $E_0^k = E_0^{k-1}$;
  end
until $\delta^k < \epsilon$;

KSBS model obtained by solving problems $(P4^*)$, $(P5^*)$, $(P6^*)$, $(P7^*)$, $(P = 8^*)$ and $(P9^*)$[13] for the network topology described in section 3.1 using Algorithm 1. The E_{budget} has been fixed to 50% and L_{max} has been varied in the interval $[500, 3000]$ ms. Fig. 12.{a-f} plot the results obtained by solving the same problems when fixing L_{max} to 3000ms and varying the E_{budget} in the interval $[1.2, 20]\%$. As it can be observed from Fig. 10, relaxing the e2e packet

---

[13]Problems $P(5^*)$-$(P9^*)$ for BMAC, X-MAC, SMAC, DMAC and LMAC are given in Appendix A.
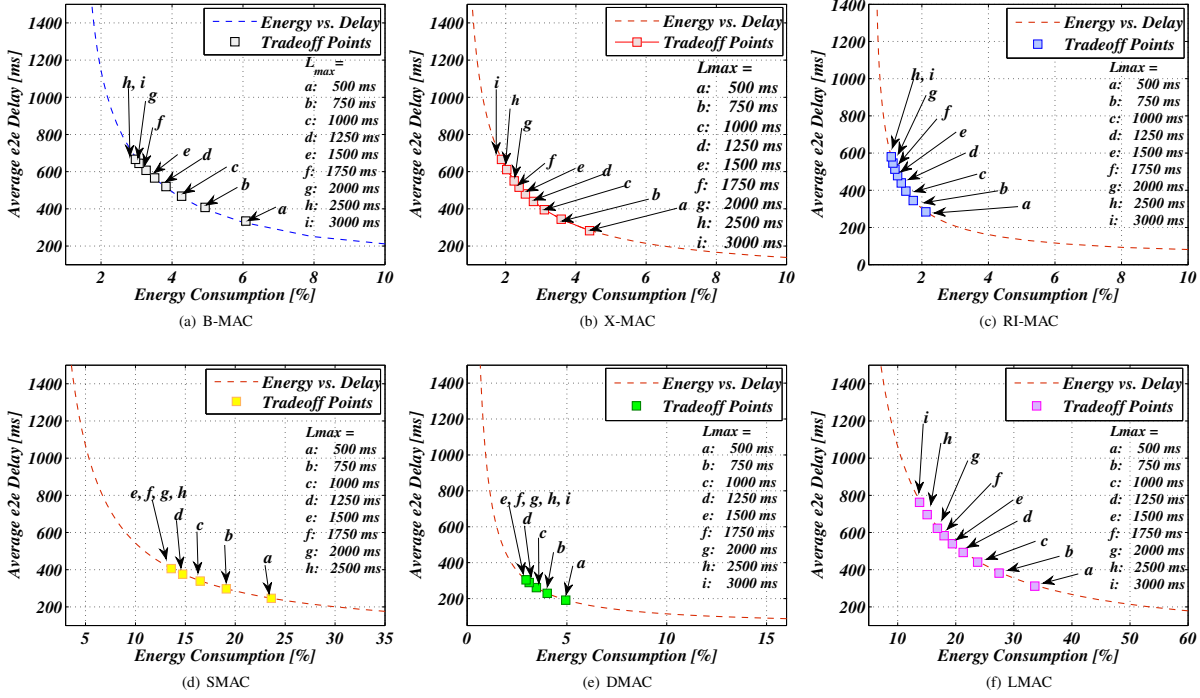
Fig. 10. Energy-Delay KSBS optimal points when fixing $E_{\text{budget}} = 50\%$ and varying the maximum e2e delay bound $L_{\text{max}} \in [500, 3000]$ms for (a) B-MAC, (b) X-MAC, (c) RI-MAC, (d) SMAC, (e) DMAC, and (f) LMAC.

Fig. 11. The obtained optimal parameters using the KSBS model: (a) the wake-up period T_w^* for B-MAC, X-MAC, and RI-MAC, and (c) the active period T_{active}^* and sleep period T_{sleep}^* for SMAC, and (d) the synchronization period T_{sync}^* for DMAC and (e) the frame size T_{frame}^* for DMAC and LMAC as trade-off points for different values of maximum e2e delay bound $L_{\text{max}} \in [500, 3000]$ms.

delay bound (L_{max}) will result in a new application requirement configuration for every protocol, and the game leads to an agreement at different trade-off points. Decreasing the L_{max} value for a fixed E_{budget} produces a lower L_{worst} value and thus a smaller interval $[L_{\text{best}}, L_{\text{worst}}]$. This results in a shorter space for gain, e.g., trade-off points with letters a,b,c. On the other hand, when L_{max} increases, L_{worst} also increases, which produces a large interval $[L_{\text{best}}, L_{\text{worst}}]$. This gives space for a large gain improvement, e.g, trade-off points with letters g,h,i. Moreover, the energy gain is proportionally equal to the latency gain. Thus, any energy gain is constrained by the L_{max}. Fig. 11.{a-d} show how the correspondent optimal parameters increase as L_{max} increases, i.e., increasing L_{max} allows low duty-cycles.

A similar behavior occurs when L_{max} is fixed to 3000ms and E_{budget} is varied, Fig. 12.{a-f}. Small values of E_{budget} produces small values of E_{worst}, and then short intervals of $[E_{\text{best}}, E_{\text{worst}}]$. This results in limited gains, e.g., trade-off points with letters a,b,c. Large values of E_{budget} produce large values of E_{worst} and then long intervals of $[E_{\text{best}}, E_{\text{worst}}]$.
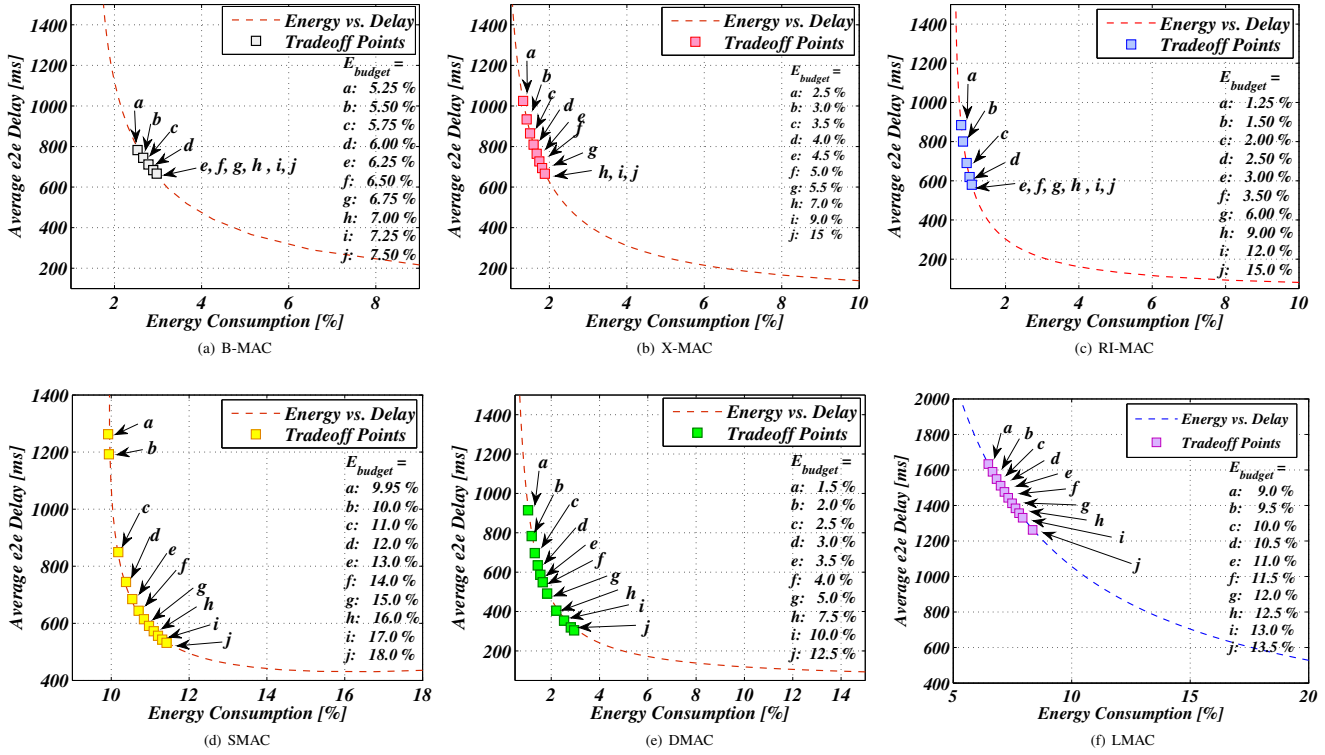
Fig. 12.   Energy-Delay KSBS optimal points when fixing $L_{max} = 3000\ ms$ and varying the maximum energy budget $E_{\text{budget}} \in [1.2, 20]\%$ for (a) B-MAC, (b) X-MAC, (c) RI-MAC, (d) SMAC, (e) DMAC, and (f) LMAC.
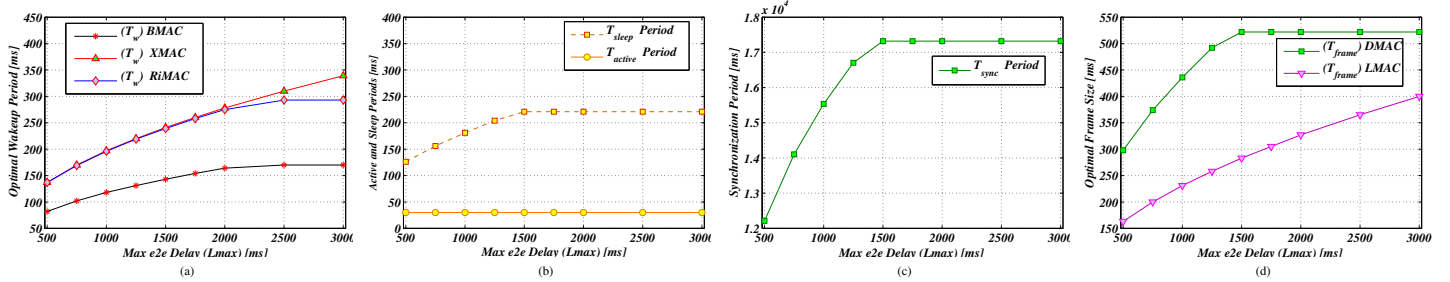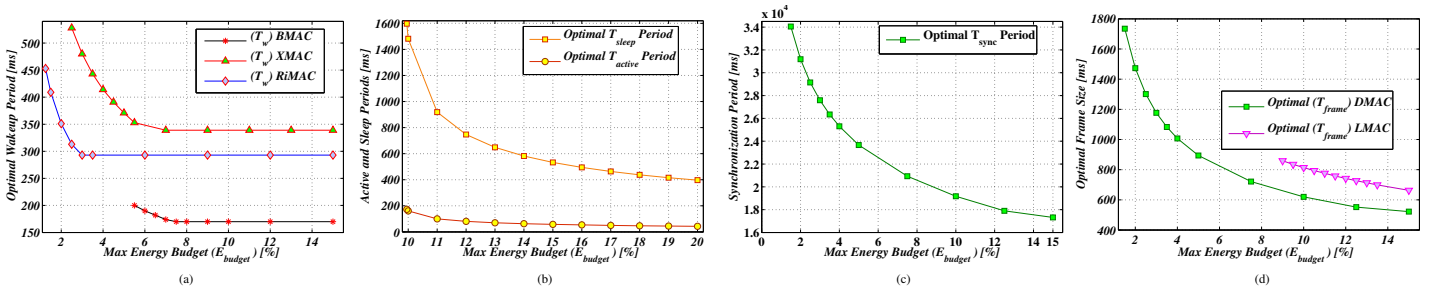

Fig. 13.   The obtained optimal parameters using the KSBS model: (a) the wake-up period T_w^* for B-MAC, X-MAC, and RI-MAC, and (c) the active period T_{active}^* and sleep period T_{sleep}^* for SMAC, and (d) the synchronization period T_{sync}^* for DMAC and (e) the frame size T_{frame}^* for DMAC and LMAC as trade-off points for different values of maximum energy budget $E_{\text{budget}} \in [1.2, 20]\%$.

This results in larger gains, e.g., trade-off points with letters h,i,j. Fig. 13.{a-d} show how the correspondent optimal parameters decrease as E_{budget} increases. It is to say, increasing energy budgets allows for high duty-cycles designs.

4.2.5. **NBS and KSBS gains**. The gain obtained by energy consumption and e2e delay players in the KSBS game is plotted in Fig 14. The gain is expressed in Eq.(3). It illustrates how much each player has proportionally reduced its cost, starting from its worst case value until each one achieves the agreement point. Fig. 14.(a) depicts the gains of different MAC protocols when the maximum e2e delay bound, L_{max}, is varied between $500ms$ and $3000ms$, and when E_{budget} is fixed to 50%. Fig. 14.(b) depicts the gains obtained when the maximum energy budget, E_{budget} is varied between 5% and 15%, and L_{max} is fixed to $3000ms$. The figures show increasing gain with the delay bound
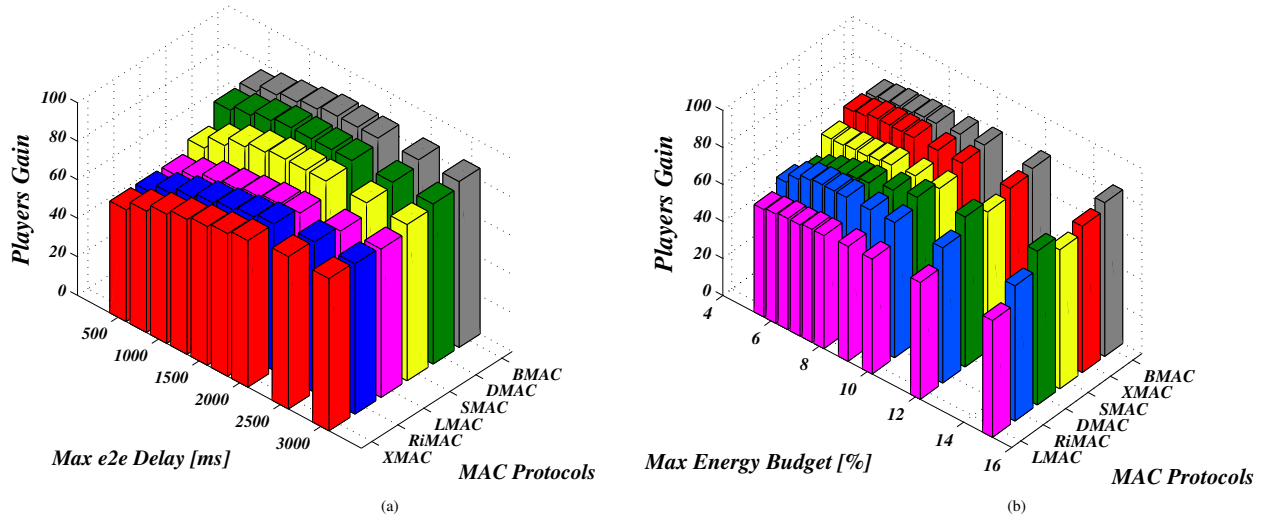
Fig. 14.   The gain obtained by each player to achieve the KSBS solution for: (a) different values of the maximum e2e delay bound L_{max} and E_{budget}=50%, (b) different values of the maximum energy budget E_{budget} and L_{max}=3000ms.

increase, as well as the energy budget increase. The different MAC protocols behave differently as the application requirements, L_{max} and E_{budget}, are modified. In one hand, all MAC protocols have obtained similar gains for large values of L_{max} (larger than $1500ms$). While for lower values of L_{max} ($500ms$ to $1500ms$), B-MAC, DMAC and SMAC have achieved better gains compared to X-MAC, RI-MAC and LMAC. In fact, to satisfy the constraint on the small values of L_{max} (in problem $(P1)$), each protocol attempts to minimize the energy while ensuring the e2e delay bound. For instance, following the design of B-MAC, SMAC, and DMAC respectively, this constraint forces B-MAC to use small wake-up periods since its one-hop delay spans for a whole wake-up period (the size of the long preamble). The same constraint requires that SMAC increases its active period and DMAC increases the size of active slots since packets in intermediate nodes can only be forwarded during these periods. Otherwise, these packets will be delayed to the next operating cycle. As a result, these protocols increase their energy consumption. The application of the game theory optimization framework enables the energy player to reduce considerably its cost value at the cost of of relaxing the e2e delay cost. On the other hand, most of the MAC protocols, except LMAC, have achieved similar incremental gains when the E_{budget} of each node is relaxed. This is because the solution of problem $(P2)$ will give lower values of L_{best} and higher values of E_{worst}. However, following the design of LMAC protocol, each node chooses randomly one transmission slot and communicates it to its neighbors, while it must perform carrier sensing in each slot to check for eventual incoming data packets. This makes the duty-cycle of nodes to be high (over 9% in Fig.6) compared to other MAC, and varying E_{budget} in 5% to 15% will not allow for better e2e delay reduction L_{best} due to the constraint on the maximum energy budget (see $(P9')$). While the solution of problem $(P1)$ will make all MAC to use longer sleep periods (due to the increase in the wake-up period for B-MAC, X-MAC, and RI-MAC, the sleep period for SMAC, the frame period for DMAC and LMAC), which results in low E_{best} and high L_{worst}. Consequently, the application of the game theory optimization framework will enable the delay player to reduce much from its cost value at the cost of relaxing the energy cost.

   From the comparison between NBS and KSBS models, the gains obtained by the energy consumption player and the e2e delay using the KSBS and the NBS models for all the duty-cycled MAC protocols are plotted in Fig.15.(a-b) for different maximum delay and budget thresholds, namely $(15\%, 1200ms)$ and $(30\%, 3000ms)$. It is clear that the use of each model depends on the assumption on whether the players follow the independent axiom defined by Nash, or the monotonicity axiom defined by Kalai-Smorodinsky. We think that in energy-delay game, there is no preference for any of the players, and thus the proportional solution of the KSBS model gives a definition of fairness between players.
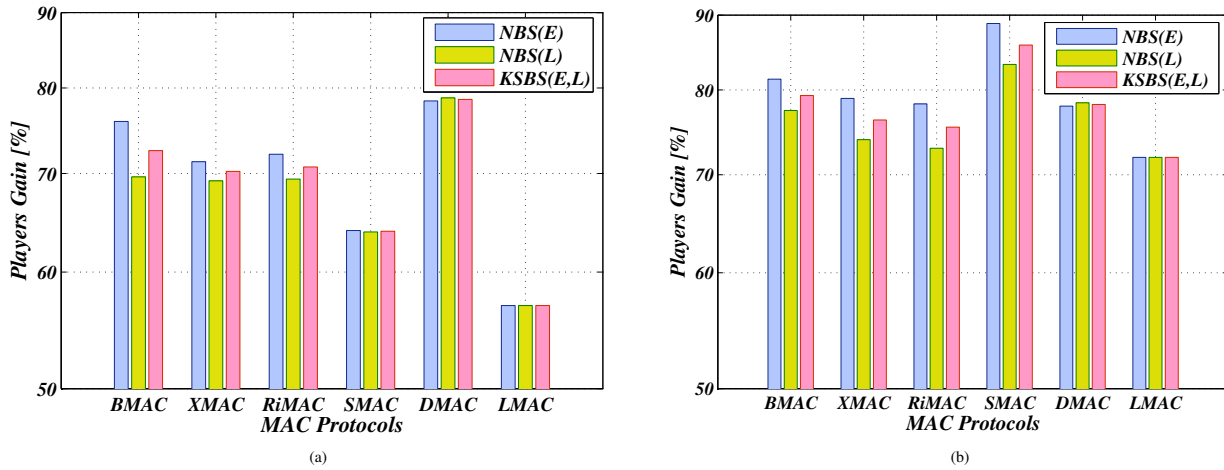
Fig. 15. Obtained gains by each player for the NBS and KSBS models. a) E_{budget}=15% and L_{max}=1200ms, b) E_{budget}=30% and L_{max}=3000ms.

Table IV. The existence of trade-off solution between energy and delay of MAC protocols for different sampling rates.

	B-MAC	X-MAC	RI-MAC	SMAC	DMAC	LMAC
F_s Threshold (pkts/min)	1/2	1/10	1/20	2	1/5	2

4.3. KSBS optimal points vs. Sampling Rate (F_s)

Another important factor in the proposed optimization framework is the sampling rate, which is an application dependent parameter. Note that our traffic model is designed for very low data rate sampling applications. A sampling rate above a given threshold may not result in the desired trade-off between energy consumption and the e2e delay objectives. In the following, we study how the sampling rate impacts the optimization model. Table IV shows the obtained sampling rate thresholds for different MAC protocols, when the maximum e2e delay bound and the maximum energy budget are $L_{max} = 5000ms$ and $E_{budget} = 50\%$, respectively. The consider network contains 200 nodes, where the node density is fixed to C=8, and the network depths has been set to D=5 levels. The thresholds are mainly defined by the bottleneck constrains for every protocol. From the table, the obtained thresholds were 1 $pkt/2min$ for B-MAC, 1 $pkt/10min$ for X-MAC, 1 $pkt/20min$ for RI-MAC, 2 $pkts/min$ for SMAC, 1 $pkts/5min$ for DMAC, and 2 $pkts/min$ for LMAC. The results reveal that each MAC protocol has its own maximum sampling frequency defined by its bottleneck constrain over which the trade-off point cannot be reached for the given application requirements configuration.

On the other hand, playing with the sampling rate under the maximum threshold can lead to different trade-off points between the two performances metrics (energy and delay), since decreasing the sampling rate will consume less energy, which gives more optimization space for both players. Given the same application requirements configuration (L_{max}=5000ms and $E_{budget} = 50\%$), we run the optimization framework for each MAC protocol and we varied the sampling rate $F_s \in [1/10, 1/60]$pkts/min. Fig. 16.a and Fig. 16.b plot the energy consumption E^* and the average e2e delay L^* corresponding to each obtained trade-off point for different sampling rates. The results show that decreasing the sampling frequency gives different behavior depending on the intrinsic definition of the MAC protocols. For example, SMAC and LMAC obtain flat responses with respect the sampling frequency. In fact, given that the traffic rate threshold of SMAC and LMAC is 2 pkts/min (Table IV) and varying the traffic in $[1/10, 1/60]$ pkts/min has less effect on the duty-cycle of SMAC and LMAC, which is mainly impacted by the active period duration and the carrier sensing in each slot for SMAC (see appendix Sec.A.1.2) and LMAC (see appendix Sec.A.1.3), respectively. As a consequence, decreasing the traffic rate under $1/10$ pkts/min will not enable for more energy saving and thus no more optimization space for SMAC and LMAC under these configurations. While reducing traffic rate from $1/10$ to $1/60$ pkts/min has allowed for more energy saving, in transmitting/receiving modes (E_{tx} and E_{rx} respectively), for other MAC (B-MAC,
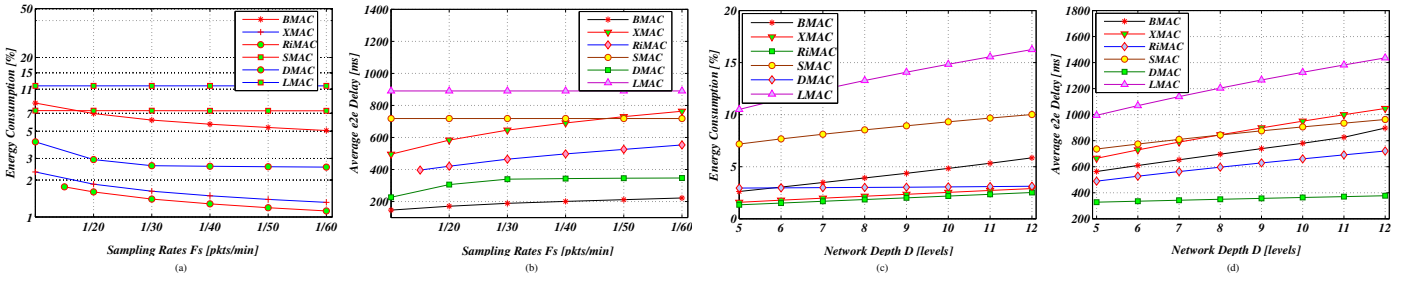
Fig. 16. The KSBS optimal points of (a) Energy Consumption E vs. different sampling rates F_s, (b) e2e Delay L vs. different sampling rates F_s, (c) Energy Consumption E vs. different network depths D, (d) e2e Delay L vs. different network depths D for each MAC protocol.

X-MAC, RI-MAC and DMAC) protocols. This will give more optimization space for both players and consequently will result in different tradeoff points for these protocols (Fig. 16{a-b}).

4.4. Scalability of the Solution

The network depth and the node density (D and C in table I) are two other system parameters that can affect the system optimization, and they have direct relation with the scalability of the solution. In this paper, a 2-players game is defined, which is independent of the number of nodes and the topology of the network. Topology and number of nodes impact in the delay and energy equations, and mainly the bottleneck conditions of each protocol that fix the applicability of the model. To show the effect of the network topology, we have run the optimization framework for every MAC protocol under several network depths (D). The sampling rate has been fixed to $1\ pkt/30min$, and the application requirements L_{max} and E_{budget} to be 5000ms and 40%, respectively. D has been varied from 5 to 12 levels, which results in a networks size from 200 to 1152 nodes, and the density has also been fixed to C=8. Fig. 16.c and Fig. 16.d plot the energy consumption E^* and the average e2e delay L^*, corresponding to each trade-off point obtained for the different network depths. It can be observed from the figures that both players increase their cost functions to reach the desired trade-off. This is due to the fact that increasing the network depth increases the e2e average delay, since this yields routes with more hops, and it increases the amount of packets that the nodes at the first level have to relay. In other words, changing the topology, the aggregated sampling rate is increased and the e2e delay equations are modified, but this has no effect on the convergence or complexity of the game model. The most interesting is that the framework is still able to find the trade-off points even in large networks, which makes the model appropriate for arbitrary network sizes.

5. RESULTS VALIDATION

Protocol optimization and the optimal parameters that permit to achieve trade-off between the energy consumption and the average e2e delay performance metrics are evaluated in this section through an extensive set of simulations under the traffic, network, and radio models defined in section 3. The objective is to show how close are the analytical results from the simulations. All the optimized MAC protocols are implemented using TinyOS operating system [Levis et al. 2004], and the evaluation is carried out with simulations using Tossim [Levis et al. 2003] (the TinyOS simulator). Very low data rate is considered ($[0.01, 10]pkts/min$). The simulated network topology is depicted in Fig. 2, where nodes are uniformly deployed with unit disk of, C=8, (the density) and the number of levels in the network is set to, D=5 (the network depth). Table I and Table II sketch all the set-up configuration parameters of the network and traffic models, as well as the typical parameter values for the CC2420 radio. In the simulation, a phase of neighbor discovery and tree construction with a minimum hop-count routes, according to the network model of Sec. 3.1, is executed before entering in sampling phase. In this phase, every node periodically generates traffic, i.e. a packet, with frequency $F_s = 1pkt/10min$. A node either transmits its local data or forwards data packets being received to the base station located t the center of the network. The validation has been performed using the NBS model described in Sec.4.2.3[14]. The network is layered into levels and packets are forwarded from outer to inner levels until reaching the

---

[14]Although the NBS has been used for simplicity to validate numerical results by simulations, this will validate implicitly the KSBS model since it is achieved by iterative NBS.
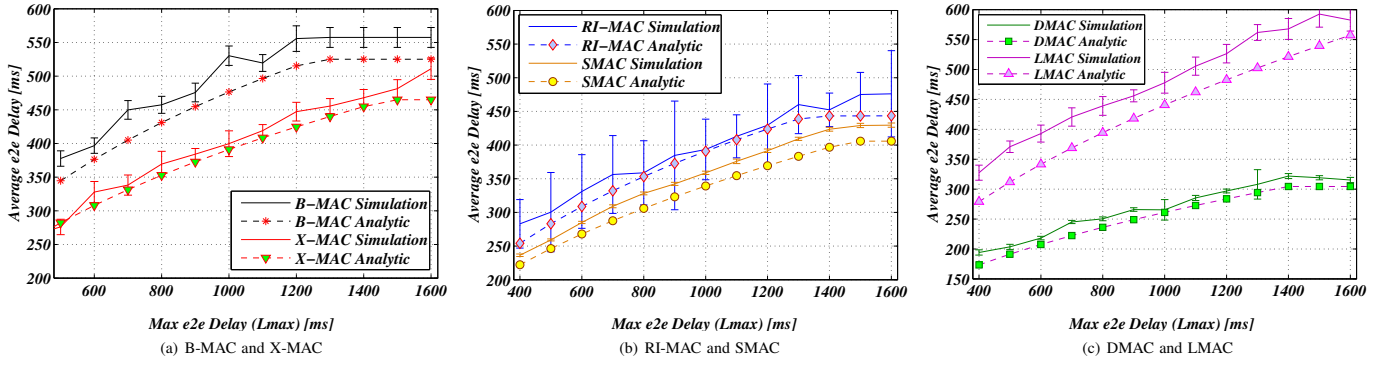
Fig. 17. Simulation results of average e2e delay for different values of the maximum latency L_{max} of the application for (a) B-MAC and X-MAC, (b) RI-MAC and SMAC, (c) DMAC and LMAC.

sink node. Every simulation is repeated at least 33 times and each point in the following curves comes from the average result of all experiments, where bars are represented with 95% of confidence interval.

5.1. Average e2e Delay vs. Max Delay Bound (L_{max})

As the framework has two different application specific requirements inputs, the maximum e2e delay bound and the maximum energy budget, we have applied and tested the proposed framework for different configurations of these application requirements. In the first scenario, the MAC protocols are optimized where the trade-off points for different values of the maximum e2e delay bound were found. The protocols are then tuned and simulated with the optimal optimization parameters obtained by solving problems $(P4^*)$, $(P5^*)$, $(P6^*)$, $(P7^*)$, $(P8^*)$, and $(P9^*)$ when fixing the maximum energy budget E_{budget} to 50% and varying the maximum e2e delay bound of the application $L_{max} \in [500, 1600]ms$. The average e2e delay of the network is measured and plotted in Fig. 17.a for B-MAC and X-MAC, and in Fig. 17.b for RI-MAC and SMAC, and in Fig. 17.c for B-MAC and X-MAC protocols. It can be observed from the figures that the average e2e delay of the network increases by increasing the maximum delay bound L_{max} of the application. The figures show that all protocols have raised their average e2e delay: 47% for B-MAC, 95% for X-MAC, 68% for RI-MAC, 81% for SMAC, 62% for DMAC, and 77% for LMAC when relaxing the maximum e2e delay bound L_{max} from $500ms$ to $1600ms$. This can be explained by the increase in the wake-up period T_w, the sleep period T_{sleep}, and the frame size T_{frame} parameters to reduce energy consumption in favor of the energy efficiency player. In fact, the e2e delay is considered as the sum of the one-hop delay along the path toward the sink. Since each node must wait half the wake-up time of its receiver in average before sending the data packet, the increase of this parameter (the wake-up time) when relaxing the e2e delay bound yields the increase in the one-hop delay and thus in the average e2e delay. It is clear from the figures that the obtained e2e delay in simulation are similar and theirs plots have the same shapes as those obtained analytically. The closest result have been obtained for X-MAC, SMAC and DMAC. The e2e delay B-MAC has some fluctuations that can be due to the randomization of the wake-up times of nodes. The e2e delay of RI-MAC is more fluctuating, and this can be explained by the losses in beacon transmissions that result in increased delays. Finally, LMAC has a simulation e2e delay a bit higher than the one obtained analytically. This is due to the position of transmission slots in the frame, where each node chooses randomly one slot to send its data (see Fig. 25).

5.2. Energy vs. Max Delay Bound (L_{max})

Fig. 18 shows the results for the same setting ($E_{budget} = 50\%$ and $L_{max} \in [500, 1600]ms$), but it plots the maximum energy consumption (duty-cycle) as L_{max} varies, Fig. 18.a for B-MAC and X-MAC, Fig. 18.b for RI-MAC and SMAC, and Fig. 18.c for B-MAC and X-MAC, respectively. The figures show that the measured energy consumption, which is the maximum consumed energy in the network, decreases with the increase in the maximum delay bound L_{max} of the application. This is due to the increase in the wake-up period T_w, the sleep period T_{sleep}, and the frame size T_{frame}. All protocols have achieved clear reduction in their respective duty-cycles (over 34%), which reduces energy consumption. Referring to Table V that sketches the obtained results, the reduction was -34% for B-MAC, -43% for
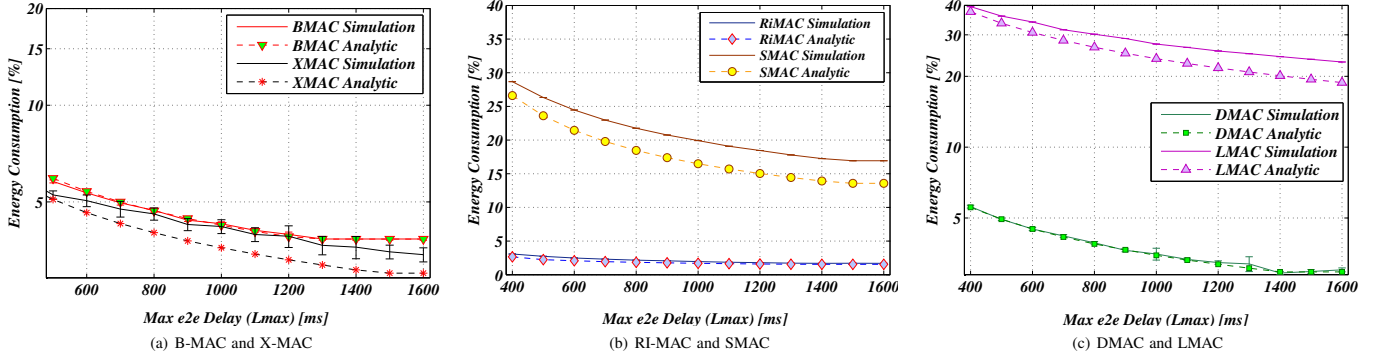
Fig. 18. Simulation results of energy consumption for different values of the maximum latency $L_{\max}$ of the application for (a) B-MAC and X-MAC, (b) RI-MAC and SMAC, (c) DMAC and LMAC.

Table V. Summary of the obtained gain and the paid cost of each player per MAC protocol for different application requirement configurations.

	B-MAC	**X-MAC**	**RI-MAC**	**SMAC**	**DMAC**	**LMAC**
$L_{\max} \in [500, 1600]$ms						
Avg e2e Delay L	+47%	+95%	+68%	+81%	+62%	+77%
Max Energy E	−34%	−43%	−45%	−41%	−46%	−42%
$E_{\mathrm{budget}} \in [1.25, 20]$%						
Avg e2e Delay L	−32%	−44%	−41%	−54%	−39%	−18%
Max Energy E	+42%	+65%	+39%	+5%	+57%	+18%

X-MAC, -45% for RI-MAC, -41% for SMAC, -46% for DMAC, and -42% for LMAC. Furthermore, the measured energy consumption is quiet close to the analytical results for all the protocols.

5.3. Average e2e Delay vs. Max Energy Budget (E_{budget})

Now, the maximum e2e delay bound, $L_{\max}$, is fixed to $1600ms$, and the maximum energy budget devoted to each node in the network, E_{budget}, is varied in the interval $[1.25, 15]\%$. The MAC protocols are configured with the optimal values obtained by solving problems ($\boldsymbol{P4^*}$), ($\boldsymbol{P5^*}$), ($\boldsymbol{P6^*}$), ($\boldsymbol{P7^*}$), ($\boldsymbol{P8^*}$), and ($\boldsymbol{P9^*}$). The resulted average e2e delay in the network is measured and plotted in Fig. 19.{a-e} for all protocols. It can be observed from the figures that the average e2e delay of the network decreases when allowing nodes to increase their duty-cycle through relaxing the maximum energy budget of the application (E_{budget}). This is achieved by decreasing the wake-up period T_w, the sleep period T_{sleep}, and the frame size T_{frame}, such that each node can quickly reach its receiver, which reduces the one-hop delay. The cost of increasing the duty-cycle by the energy player goes in favor of the e2e delay player. The reduction in the e2e delay of each protocol when increasing the maximum allowed duty-cycle is shown in Table V, which varies between -18% and -54%. The measured e2e delay from the simulations are close to the analytical e2e delay for all the protocols. It can be observed that the protocols with the most convergent results are X-MAC, SMAC and DMAC. Lower active periods T_{active} cause some packets to be delayed, which explains the large average e2e delay for SMAC (for low values of E_{budget}). The e2e delays of B-MAC and RI-MAC have some fluctuations for the reasons explained above. Whereas, the the e2e delay of LMAC is a bit over that of the analytical model. This may be explained by the randomness of the reception slots position in the transmission frame.

5.4. Energy vs. Max Energy Budget (E_{budget})

Finally, the maximum consumed energy in the network is also measured during the simulations and the results are depicted in Fig. 20.{a-e}. As expected from the analytical results and confirmed by the simulations, the figures show
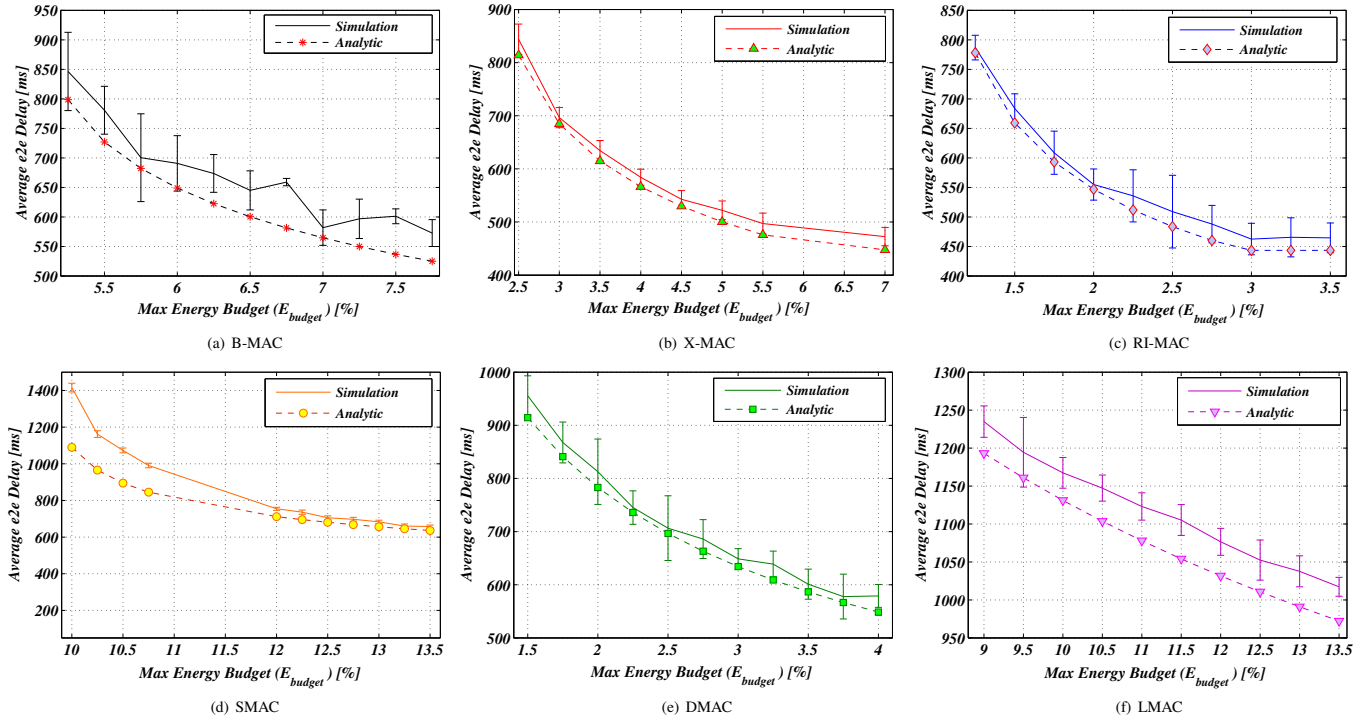
Fig. 19. Simulation results of average e2e delay for different values of the maximum energy budget E_{budget} of the application for (a) B-MAC, (b) X-MAC, (c) RI-MAC, (d) SMAC, (e) DMAC, and (f) LMAC.

that the nodes increase their duty-cycle (between $+5\%$ and $+65\%$) when they are allocated more energy budget E_{budget}. This lost in energy saving goes in favor of the e2e delay player as seen in section 5.3. This is due to the decrease in the wake-up period, T_w, the sleep period, T_{sleep}, and the frame size, T_{frame}. The simulations show that the protocols can achieve the desired trade-off and provide results close to the analytically calculated energy. All the obtained gains and paid costs of each player for different configurations of the application requirements are summarized in Table V for comparison. Although the gains and the costs between the energy and the delay players were not the same, there is a clear trading between them to achieve the desired proportional optimization. These results reveal that tuning the MAC protocols with optimized parameters gives better balance between energy efficiency and e2e delay performances.

6. CONCLUSION

The problem of optimizing system performance subject to conflicting needs has been targeted in this paper, where a general optimization framework for duty-cycled MAC protocols in low data rate wireless networks is proposed. The energy-delay trade-off have been investigated from a game theory perspective and showed how to appropriately achieve a fair trade-off between them. Different from multi-objective optimization methods, which either optimize one performance objective and treat the others as constraints of the problem, or aggregate all performance metrics into a single objective function with different weights and from the cooperative games which use the nodes as the players, we have treated the two objectives as two players in a cooperative game. We accordingly proposed an appealing method to find the fair performances trade-off between energy consumption and the e2e delay, where each of the two players reduces its cost function that is minimized with the same order of magnitude, which makes the solution equally fair. First, the players negotiate with each other in order to achieve an agreement under the Nash Bargaining equilibrium. Then, an iterative algorithm is proposed that uses the Nash Bargaining model in a repeated game to find the solution under the Kalai-Smorodinsky Bargaining model. The framework is scalable with the number of nodes and is general enough to be applied to any wireless MAC protocol, notably those proposed for WSN. The framework has been applied to determine the optimal MAC parameters of six WSN duty-cycled MAC protocols, B-MAC, X-MAC, RI-MAC, SMAC, DMAC, and LMAC, and then to achieve a fair energy-delay trade-off in long run according to the
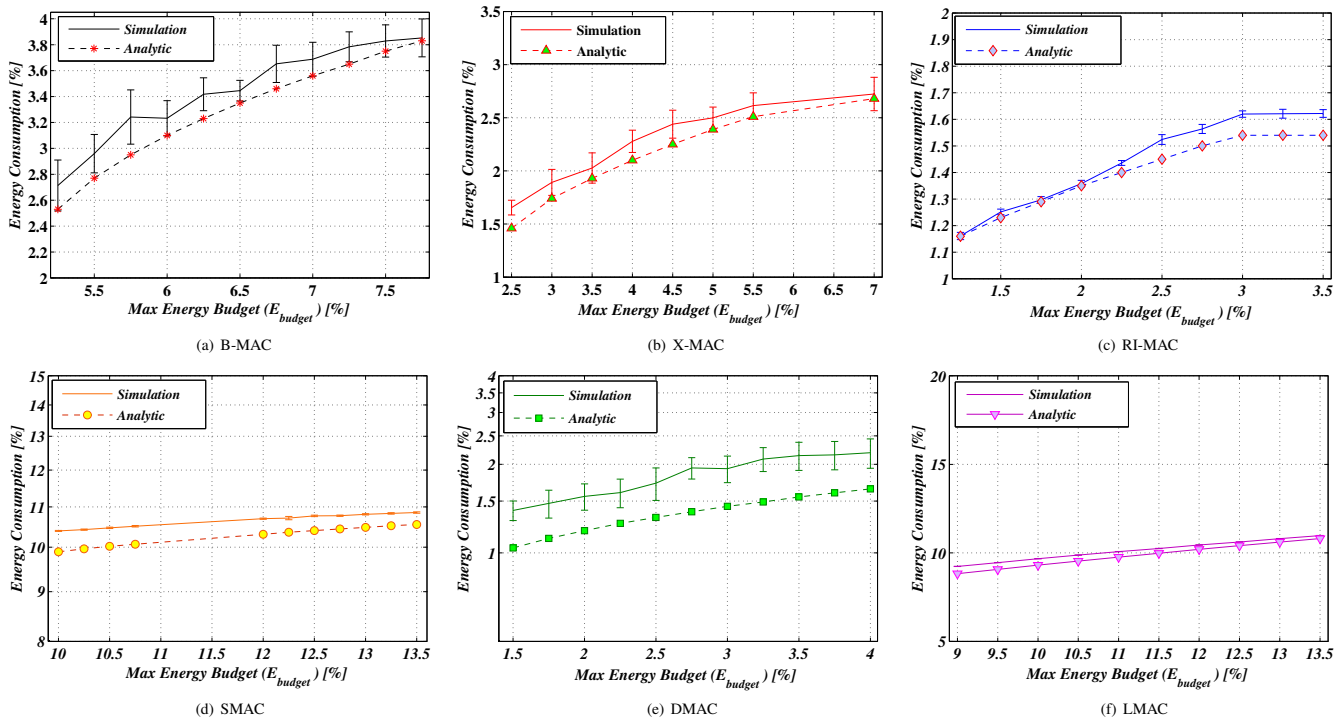
Fig. 20. Simulation results of energy consumption for different values of the maximum energy budget E_{budget} of the application for (a) B-MAC, (b) X-MAC, (c) RI-MAC, (d) SMAC, (e) DMAC, and (f) LMAC.

application requirements. Further, the optimization results have been supported by extensive simulations where the MAC protocols are implemented and tuned with the corresponding optimal parameters. The energy consumption and the average e2e delay were measured and compared to the analytical results. We found that when tuning the duty-cycled MAC protocols with the optimal parameters, they map the obtained trade-off performances and confirm the effectiveness of the proposed framework. The impact of the sampling frequency was also analyzed, and we showed that the framework is able to find the maximum sampling rate that each protocol must not violate in order to achieve the target trade-off for a given application requirement. Finally, to demonstrate the capability of the framework to run over large networks, the optimization of different MAC protocols has been tested under increasing values of the network size. This makes it scalable and suitable for arbitrary network sizes.

Although the proposed framework allows to optimize MAC protocols and find the fair trade-off between energy consumption and the e2e delay objectives, there are some factors that have been simplified by the traffic model considered, e.g, the absence of interference and retransmissions. This hides some features of the real conditions of wireless environment that are not captured by this model. We believe that these simplifications are somehow reasonable due to the very low data rate nature of the targeted applications, where the tight traffic minimizes the impact of interferences and retransmissions due to congestion and collisions. We are working to extend this model to capture these factors and to include other kind of traffic models such as Poisson arrivals.

ACKNOWLEDGMENTS

REFERENCES

2010. *CC2420, Single-Chip 2.4 GHz RF Transceiver, Texas Instruments*. http://focus.ti.com/lit/ds/symlink/cc2420.pdf

2011. *The MAC Alphabet Soup served in Wireless Sensor Networks*. http://www.st.ewi.tudelft.nl/~koen/MACsoup/taxonomy

Andrea Abrardo, Lapo Balucanti, and Alessandro Mecocci. 2013. A Game Theory Distributed Approach for Energy Optimization in WSNs. *ACM Trans. Sen. Netw.* 9, 4, Article 44 (July 2013), 22 pages. DOI:http://dx.doi.org/10.1145/2489253.2489261

F. Afghah, M. Costa, A. Razi, A. Abedi, and A. Ephremides. 2013. A reputation-based Stackelberg game approach for spectrum sharing with cognitive cooperation. In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*. 3287–3292.

Tarek AlSkaif, Manel Guerrero Zapata, and Boris Bellalta. 2015. Game theory for energy efficiency in Wireless Sensor Networks: Latest trends. *Journal of Network and Computer Applications* 54 (2015), 33 – 61.

G. Bacci, L. Sanguinetti, M. Luise, and H. V. Poor. 2013. A Game-Theoretic Approach for Energy-Efficient Contention-Based Synchronization in OFDMA Systems. *IEEE Transactions on Signal Processing* 61 (March 2013), 1258–1271.

P.V. (Sundar) Balakrishnan, Juan Camilo Gmez, and Rakesh V. Vohra. 2011. The Tempered Aspirations solution for bargaining problems with a reference point. *Mathematical Social Sciences* 62, 3 (2011), 144 – 150. DOI:http://dx.doi.org/10.1016/j.mathsocsci.2011.09.003

Saeed Bastani, Bjorn Landfeldt, Christian Rohner, and Per Gunningberg. 2012. A Social Node Model for Realising Information Dissemination Strategies in Delay Tolerant Networks. In *ACM MSWiM*. ACM, New York, NY, USA, 79–88.

HP Benson. 2004. On the global optimization of sums of linear fractional functions over a convex set. *Journal of Optimization Theory and Applications* 121, 1 (2004), 19–39.

S. Boyd and L. Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press.

Michael Buettner, Gary V. Yee, Eric Anderson, and Richard Han. 2006. X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks. In *ACM SenSys*. 307–320.

Matteo Ceriotti, Michele Corrà, Leandro D'Orazio, Roberto Doriguzzi, Daniele Facchin, Stefan Guna, Gian Paolo Jesi, Renato Lo Cigno, Luca Mottola, Amy L. Murphy, Massimo Pescalli, Gian Pietro Picco, Denis Pregnolato, and Carloalberto Torghele. 2011. Is there light at the ends of the tunnel? Wireless sensor networks for adaptive lighting in road tunnels. In *IPSN*. 187–198.

Jie Chen and A.L. Swindlehurst. 2012. Applying Bargaining Solutions to Resource Allocation in Multiuser MIMO-OFDMA Broadcast Systems. *Selected Topics in Signal Processing, IEEE Journal of* 6, 2 (April 2012), 127–139.

Xiaoyu Chu and Harish Sethu. 2015. Cooperative Topology Control with Adaptation for improved lifetime in wireless sensor networks. *Ad Hoc Networks* 30 (2015), 99 – 114.

Messaoud Doudou, Djamel Djenouri, and Nadjib Badache. 2013. Survey on Latency Issues of Asynchronous MAC Protocols in Delay-Sensitive Wireless Sensor Networks. *IEEE Communications Surveys and Tutorials* 15, 2 (2013), 528–550.

Adam Dunkels. 2011. *The ContikiMAC Radio Duty Cycling Protocol*. Technical Report T2011:13. Swedish Institute of Computer Science. http://dunkels.com/adam/dunkels11contikimac.pdf

Prabal Dutta, Stephen Dawson-Haggerty, Yin Chen, Chieh-Jan Mike Liang, and Andreas Terzis. 2012. A-MAC: A Versatile and Efficient Receiver-initiated Link Layer for Low-power Wireless. *ACM Trans. Sen. Netw.* 8, 4 (Sept. 2012), 30:1–30:29.

CA Floudas and V Visweswaran. 1993. Primal-relaxed dual global optimization approach. *Journal of Optimization Theory and Applications* 78, 2 (1993), 187–225.

A El Gamal, James Mammen, Bharat Prabhakar, and Devavrat Shah. 2004. Throughput-delay trade-off in wireless networks. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, Vol. 1. IEEE.

A. Ghasemi and K. Faez. 2008. A nash power-aware MAC game for ad hoc wireless networks. In *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*. 1–5.

Matthias Grossglauser and David Tse. 2001. Mobility increases the capacity of ad-hoc wireless networks. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, Vol. 3. IEEE, 1360–1369.

Piyush Gupta and Panganmala R Kumar. 2000. The capacity of wireless networks. *Information Theory, IEEE Transactions on* 46, 2 (2000), 388–404.

Zhu Han, Dusit Niyato, Walid Saad, Tamer Baar, and Are Hjrungnes. 2011. *Game Theory in Wireless and Communication Networks*. Cambridge University Press. Cambridge Books Online.

Y. Hayel, S. Trajanovski, E. Altman, Huijuan Wang, and P. Van Mieghem. 2014. Complete game-theoretic characterization of SIS epidemics protection strategies. In *IEEE 53rd Annual Conference on Decision and Control (CDC)*. 1179–1184.

Jason L. Hill and David E. Culler. 2002. Mica: A Wireless Platform for Deeply Embedded Networks. *IEEE Micro* 22, 6 (Nov. 2002), 12–24.

Ehud Kalai and Meir Smorodinsky. 1975. Other Solutions to Nash's Bargaining Problem. *Econometrica* 43, 3 (1975), pp. 513–518. http://www.jstor.org/stable/1914280

F. Kelly. 1997. Charging and rate control for elastic traffic. *Europ. Trans. Telecommun.* 8, 1 (1997), 33–37.

Eunji Kim, Hyunggon Park, and P. Frossard. 2012. Low complexity iterative multimedia resource allocation based on game theoretic approach. In *Circuits and Systems (ISCAS), 2012 IEEE International Symposium on*. 1099–1102.

K. Langendoen and A. Meier. 2010. Analyzing MAC Protocols for Low Data-Rate Applications. *ACM Transactions on Sensor Networks TOSN* 7, 2 (2010).

W.L. Leow and H. Pishro-Nik. 2007. Delay and Energy Tradeoff in Multi-State Wireless Sensor Networks. In *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*. 1028–1032.

Philip Levis, Nelson Lee, Matt Welsh, and David Culler. 2003. TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*. ACM, New York, NY, USA, 126–137.

Philip Levis, Sam Madden, Joseph Polastre, Robert Szewczyk, Alec Woo, David Gay, Jason Hill, Matt Welsh, Eric Brewer, and David Culler. 2004. TinyOS: An operating system for sensor networks. In *in Ambient Intelligence*. Springer Verlag.

Gang Lu, Bhaskar Krishnamachari, and Cauligi S. Raghavendra. 2007. An adaptive energy-efficient and low-latency MAC for tree-based data gathering in sensor networks. *Wireless Communications and Mobile Computing* 7, 7 (2007), 863–875.

Kai Ma, Xinping Guan, Bin Zhao, and Juan Wang. 2011. A cooperation strategy based on bargaining solution in wireless sensor networks. *Journal of Control Theory and Applications* 9, 1 (2011), 121–126.

Renita Machado and Sirin Tekinay. 2008. A survey of game-theoretic approaches in wireless sensor networks. *Computer Networks* 52, 16 (2008), 3047 – 3061.

R. Mazumdar, L.G. Mason, and C. Douligeris. 1991. Fairness in network optimal flow control: optimality of product forms. *Communications, IEEE Transactions on* 39, 5 (May 1991), 775–782.

F. Meshkati, H.V. Poor, and S.C. Schwartz. 2009a. Energy Efficiency-Delay Tradeoffs in CDMA Networks: A Game-Theoretic Approach. *IEEE Transactions on Information Theory* 55, 7 (July 2009), 3220–3228.

F. Meshkati, H.V. Poor, S.C. Schwartz, and R.V. Balan. 2009b. Energy-efficient resource allocation in wireless networks with quality-of-service constraints. *IEEE Transactions on Communications* 57, 11 (Nov 2009), 3406–3414.

Mihail Mihaylov, Yann-Aël Le Borgne, Karl Tuyls, and Ann Nowé. 2011. Distributed Cooperation in Wireless Sensor Networks. In *The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 1 (AAMAS '11)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 249–256.

David Moss and Philip Levis. 2008. *BoX-MACs: Exploiting Physical and Link Layer Boundaries in Low-Power Networking*. Technical Report. Technical Report SING-08-00, Stanford University.

A. Nahir and A. Orda. 2007. The energy-delay tradeoff in wireless networks - system-wide optimization and game-theoretic perspectives. In *Tech. Report*. Technion University, Haifa, Israel.

John F Nash Jr. 1950. The bargaining problem. *Econometrica: Journal of the Econometric Society* (1950), 155–162.

M.J. Neely. 2007. Optimal Energy and Delay Tradeoffs for Multiuser Wireless Downlinks. *Information Theory, IEEE Transactions on* 53, 9 (Sept 2007), 3095–3113.

Michael J Neely and Eytan Modiano. 2005. Capacity and delay tradeoffs for ad hoc mobile networks. *Information Theory, IEEE Transactions on* 51, 6 (2005), 1917–1937.

N. Nissan, T. Roughgarden, E. Tardos, and V.V. Vazirani. 2007. *Algorithmic Game Theory*. Cambridge University Press.

Dusit Niyato and Ekram Hossain. 2008. Market-equilibrium, competitive, and cooperative pricing for spectrum sharing in cognitive radio networks: Analysis and comparison. *Wireless Communications, IEEE Transactions on* 7, 11 (2008), 4273–4283.

P. Nuggehalli, M. Sarkar, K. Kulkarni, and R.R. Rao. 2008. A Game-Theoretic Analysis of QoS in Wireless MAC. In *IEEE INFOCOM*. 1903–1911.

Jorge Ortn, Jos Ramn Gllego, and Mara Canales. 2013. Joint route selection and resource allocation in multihop wireless networks based on a game theoretic approach. *Ad Hoc Networks* 11, 8 (2013), 2203 – 2216.

K. Pandremmenou, L.P. Kondi, and K.E. Parsopoulos. 2013. Geometric Bargaining Approach for Optimizing Resource Allocation in Wireless Visual Sensor Networks. *Circuits and Systems for Video Technology, IEEE Transactions on* 23, 8 (Aug 2013), 1388–1401.

Hyunggon Park and M. van der Schaar. 2007. Bargaining Strategies for Networked Multimedia Resource Management. *Signal Processing, IEEE Transactions on* 55, 7 (July 2007), 3496–3511.

Pan Gun Park, Carlo Fischione, Alvise Bonivento, Karl Henrik Johansson, and Alberto L. Sangiovanni-Vincentelli. 2011. Breath: An Adaptive Protocol for Industrial Control Applications Using Wireless Sensor Networks. *IEEE Trans. Mob. Comput.* 10, 6 (2011), 821–838.

Yang Peng, Zi Li, Daji Qiao, and Wensheng Zhang. 2011. Delay-bounded MAC with minimal idle listening for sensor networks. In *IEEE INFOCOM*. 1314–1322.

Joseph Polastre, Jason Hill, and David Culler. 2004. Versatile Low Power Media Access for Wireless Sensor Networks. In *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*. ACM, New York, NY, USA, 95–107.

Yanjun Sun, Omer Gurewitz, and David B. Johnson. 2008. RI-MAC: A Receiver-initiated Asynchronous Duty Cycle MAC Protocol for Dynamic Traffic Loads in Wireless Sensor Networks. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems (SenSys '08)*. ACM, New York, NY, USA, 1–14.

C.D. Truong, M.A. Khan, F. Sivrikaya, and S. Albayrak. 2010. Cooperative game theoretic approach to energy-efficient coverage in wireless sensor networks. In *Networked Sensing Systems (INSS), 2010 Seventh International Conference on*. 73–76.

L.F.W. van Hoesel and P.J.M. Havinga. 2004. A Lightweight Medium Access Protocol (LMAC) for Wireless Sensor Networks: Reducing Preamble Transmissions and Transceiver State Switches. In *1st International Workshop on Networked Sensing Systems, INSS 2004*. Society of Instrument and Control Engineers (SICE), Tokio, Japan, 205–208.

Artemis C. Voulkidis, Markos P. Anastasopoulos, and Panayotis G. Cottis. 2013. Energy Efficiency in Wireless Sensor Networks: A Game-theoretic Approach Based on Coalition Formation. *ACM Trans. Sen. Netw.* 9, 4 (July 2013), 43:1–43:27.

O. Yang and W. R. Heinzelman. 2009. Modeling and Throughput Analysis for S-MAC with a Finite Queue Capacity. In *ISSNIP*. 409–414.

Wei Ye, John Heidemann, and Deborah Estrin. 2004. Medium Access Control with Coordinated, Adaptive Sleeping for Wireless Sensor Networks. *ACM/IEEE Transactions on Networking* 12, 3 (June 2004), 493–506.

W. Ye, F. Silva, and J. Heidemann. 2006. Ultra-low duty cycle MAC with scheduled channel polling. In *ACM SenSys*. 321–334.

Lei Ying, Sichao Yang, and R Srikant. 2008. Optimal delay–throughput tradeoffs in mobile ad hoc networks. *Information Theory, IEEE Transactions on* 54, 9 (2008), 4119–4143.

Ruifeng Zhang, Olivier Berder, Jean-Marie Gorce, and Olivier Sentieys. 2012. Energydelay tradeoff in wireless multihop networks with unreliable links. *Ad Hoc Networks* 10, 7 (2012), 1306 – 1321.

Xi Zhang and Jia Tang. 2013. Power-Delay Tradeoff over Wireless Networks. *Communications, IEEE Transactions on* 61, 9 (September 2013), 3673–3684.

Liqiang Zhao, Le Guo, Cong Li, and Hailin Zhang. 2009. An Energy-Efficient MAC Protocol for WSNs: Game-Theoretic Constraint Optimization with Multiple Objectives. *Wireless Sensor Network* (2009), 358–364.

Yangming Zhao, Sheng Wang, Sizhong Xu, Xiong Wang, Xiujiao Gao, and Chunming Qiao. 2013. Load balance vs energy efficiency in traffic engineering: A game Theoretical Perspective. In *INFOCOM, 2013 Proceedings IEEE*. 530–534.

Tao Zheng, Sridhar Radhakrishnan, and Venkatesh Sarangan. 2011. Modeling and performance analysis of DMAC for wireless sensor networks. In *ACM MSWiM*. 119–128.

Marco Zimmerling, Federico Ferrari, Luca Mottola, Thiemo Voigt, and Lothar Thiele. 2012. PTunes: runtime parameter adaptation for low-power MAC protocols. In *IEEE IPSN*. 173–184.

Online Appendix to:
Game Theory Framework for MAC Parameter Optimization in Energy-Delay Constrained Sensor Networks

MESSAOUD DOUDOU, CERIST Research Center, Algiers, Algeria
JOSE M. BARCELO-ORDINAS, Universitat Politecnica de Catalunya (UPC), Barcelona, Spain
DJAMEL DJENOURI, CERIST Research Center, Algiers, Algeria
JORGE GARCIA VIDAL, Universitat Politecnica de Catalunya (UPC), Barcelona, Spain
ABDELMADJID BOUABDALLAH, Université de Technologie de Compiègne (UTC), Compiègne, France
NADJIB BADACHE, University of Sciences and Technology (USTHB), Algiers, Algeria

---

A. APPLICATION TO DUTY-CYCLED MAC PROTOCOLS (CONTINUED)

In this section, the game theory framework is applied to B-MAC, X-MAC, SMAC, DMAC, and LMAC protocols where each protocol is first described through its operation modes, then the per node energy, e2e delay, and the bottleneck constraint are provided and used to build the system energy and delay model.

A.1. Protocols Description

*A.1.1. **Preamble-based.*** **B-MAC**, proposed by [Polastre et al. 2004], is considered as a canonical energy efficient MAC protocols, and the first LPL (Low Power Listening) protocol in which nodes wake-up periodically, every T_w period (1), to perform carrier sensing for T_{cs}. Referring to Fig. 21, when a node has a data packet to be sent it contends first to the channel during T_{cw}, then it transmits a long preamble to wake-up its receiver followed by the data packet, T_{data}, which spans for the transmission of the header and the payload (2). The long preamble spans for a complete wake-up period overlap with the wake-up time of the receiver. If a node overhears the preamble when polling the channel (3), it keeps its radio on until the reception of the header of data packet (4), to decide whether to receive the packet or to return back to sleep. After successful reception of the data packet, The receiver replies with an acknowledgment of duration T_{ack} and returns back to sleep (5). The main adjustable parameter that affects the energy and delay performance is mainly the wake-up period , T_w, and hence the vector parameters for B-MAC protocol is given by $X_{\text{BMAC}}=[T_w]$. In B-MAC, the energy is consumed in idle listening E_{cs}, in transmission, E_{tx}, in reception, E_{rx}, and in overhearing, E_{ovr}, modes. According to [Langendoen and Meier 2010], the per-node energy consumption and e2e packet delay, as well as the bottleneck constraints are expressed in the following (description of every term used in the formulas can be found in table I and table VI).

a) The energy of node n:

$$E^n = E_{cs}^n + E_{tx}^n + E_{rx}^n + E_{ovr}^n \tag{12}$$

where

$$E_{cs}^n = \frac{T_{cs}}{T_w},$$
$$E_{tx}^n = (T_{cs} + T_{cw}/2 + T_w + T_{\text{data}}) \, F_{\text{out}}^n,$$
$$E_{rx}^n = (T_w/2 + T_{\text{data}}) \, F_I^n,$$
$$E_{ovr}^n = (T_w/2 + T_{\text{hdr}}) \, F_B^n,$$

b) The delay of node n at level d:

$$L^n = \sum_{i=1}^{d} (T_w + \frac{T_{cw}}{2} + T_{\text{data}}) \tag{13}$$
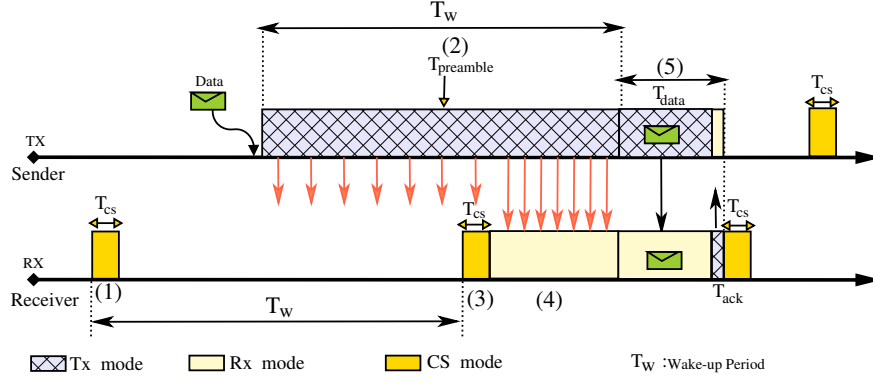
Fig. 21.   BMAC's carrier sensing, transmission, and receiving modes.

where $T_{\text{data}} = T_{\text{hdr}} + P/R + T_{\text{ack}}$.

c) The bottleneck constraints:

$$|I^0| \, E_{tx}^1 < 1/4 \tag{14}$$

From equations (12-14), the following system wide energy-delay functions are defined:

a) The network energy consumption function:

$$\boldsymbol{E}^{\text{B-MAC}} = \max_{n \in N}\left(\frac{\alpha_1}{T_w} + \alpha_2 * T_w + \alpha_3\right) \tag{15}$$

where $\alpha_1 = T_{cs}$, $\alpha_2 = (T_{cs} + T_{\text{data}})\,F_{\text{out}}^n + T_{\text{data}}\,F_I^n + T_{\text{hdr}}\,F_B^n$, and $\alpha_3 = F_{\text{out}}^n + \frac{F_I^n + F_B^n}{2}$.

b) The e2e packet delay function:

$$\boldsymbol{L}^{\text{B-MAC}} = \max_{n \in N}(\beta_1 T_w + \beta_2) \tag{16}$$

where $\beta_1 = \sum_{i=1}^d 1$ and $\beta_2 = \sum_{i=1}^d\left(\frac{T_{cw}}{2} + T_{\text{data}}\right)$.

X-MAC, proposed by [Buettner et al. 2006], is an asynchronous preamble sampling based protocol where nodes wake up periodically every T_w seconds to perform carrier sensing for, $T_{cs} + T_{al}$, (1) as depicted in Fig. 22. To send a packet, a node first contends to access the channel within the contention window T_{cw}, and it transmits a sequence of strobe preambles of duration T_{ps}, which are short packets containing the identifier of the receiver. It then listens to an acknowledgment for T_{al} (2). Strobes continue for a period sufficient to make at least one strobe overlap with a receiver wake-up (3). The receiver replies with an acknowledgment of duration T_{ack} (4) and keeps the radio on. After that, the sender transmits the data packet, T_{data}, which spans for the transmission of the header and the payload (5). Similarly to B-MAC, the main adjustable parameter that affects the energy and delay performance is mainly the wake-up period, T_w. The vector parameter for X-MAC protocol is given by $X_{\text{XMAC}} = [T_w]$. The per-node energy consumption based on the protocol operation modes, the e2e packet delay, and the bottleneck constraint are given in the following (description of every term used in the formulas can be found in table I and table VI):

a) The Energy of node n:

$$E^n = E_{cs}^n + E_{tx}^n + E_{rx}^n + E_{ovr}^n, \tag{17}$$
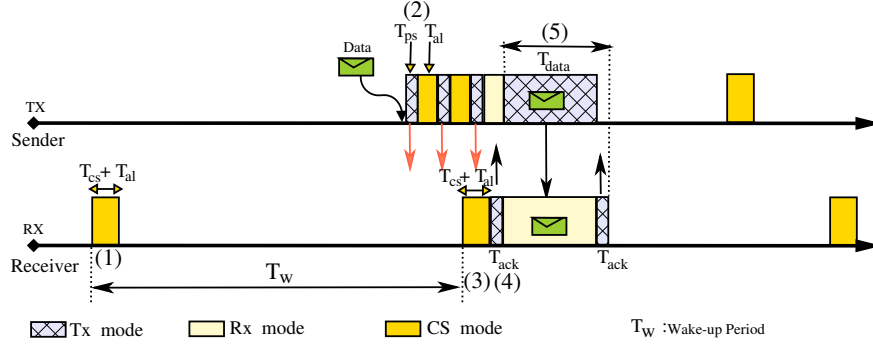
Fig. 22.   X-MAC's carrier sensing, transmission, and receiving modes.

where $E_{\text{cs}}^n = \frac{(T_{cs}+T_{al})}{T_w}$, $T_{tx} = \lceil \frac{T_w}{T_{ps}+T_{al}} \rceil \frac{T_{ps}+T_{al}}{2} + T_{ack} + T_{data}$, $E_{\text{tx}}^n = (T_{cs}+T_{al}+T_{tx})\, F_{\text{out}}^n$, $E_{rx}^n = (\frac{3}{2}T_{ps}+T_{ack}+T_{data})\, F_I^n$, and $E_{\text{ovr}}^n = (\frac{3}{2}\frac{T_{\text{tx}}}{T_w}T_{ps})\, F_B^n$.

b) The delay of node n at level d:

$$L^n = \sum_{i=1}^{d} \left(\frac{T_w}{2} + \frac{T_{cw}}{2} + T_{\text{data}} \right), \tag{18}$$

where $T_{\text{data}} = T_{\text{hdr}} + P/R + T_{\text{ack}}$.

c) The bottleneck constraint:

$$|I^0|\, E_{tx}^1 < 1/4 \tag{19}$$

From equations (17-19), the following system wide energy-delay functions are defined:

a) The network energy consumption function:

$$\boldsymbol{E}^{\text{XMAC}} = \max_{n \in N} \left(\frac{\eta_1}{T_w} + \eta_2 T_w + \eta_3 \right) \tag{20}$$

where $\alpha_1 = T_{cs} + T_{al} + \frac{3}{2}T_{ps}(\frac{T_{ps}+T_{al}}{2} + T_{ack} + T_{data})F_B^n$, $\alpha_2 = (\frac{T_{ps}+T_{al}}{2} + T_{cs} + T_{al} + T_{ack} + T_{data})F_{out}^n + (\frac{3}{2}T_{ps} + T_{ack} + T_{data})F_I^n + \frac{3}{2}T_{ps}F_B^n$, and $\alpha_3 = \frac{F_{out}^n}{2}$.

b) The e2e packet delay function:

$$\boldsymbol{L}^{\text{XMAC}} = \max_{n \in N} \left(\beta_1 T_w + \beta_2 \right) \tag{21}$$

where $\beta_1 = \sum_{i=1}^{d} 1/2$ and $\beta_2 = \sum_{i=1}^{d} (\frac{T_{cw}}{2} + T_{data})$.

*A.1.2. **Slot-based**.* **SMAC** by [Ye et al. 2004] is the baseline of slotted contention-based MAC protocols in which neighboring nodes follow the same schedule forming thus a virtual cluster. As depicted in Fig. 23, nodes are synchronized to a common slot structure of a fixed length T_{slot} (1). The slots are divided into an active period T_{active} (2), a sync phase T_{sync} (3), and a sleep phase T_{sleep} (4). Nodes broadcast SYNC packets to keep the network synchronized (5). The

Table VI. Different MAC Symbols used in Energy & Delay Equations.

MAC	Parameter & Description	Values
B-MAC	T_w: Wake-up period [ms] T_{cw}: Contention window size [ms] T_{hdr}, T_{ack}: Pkt header & Ack duration [ms]	T_w^* $15 times 0.62$ $\frac{9}{R}, \frac{9+L_{pbl}}{R}$
X-MAC	T_w: Wake-up period [ms] T_{al}: Early-Ack listen period [ms] T_{ps}: Strobe preamble duration [ms] T_{cw}: Contention window size [ms] T_{hdr}, T_{ack}: Pkt header & Ack duration [ms]	T_w^* 0.95 $\frac{5+L_{pbl}}{R}$ $15 times 0.62$ $\frac{9+L_{pbl}}{R}$
SMAC	T_{slot}: Slot duration [ms] T_{sync}: Synchronization period [ms] T_{active}: Active period [ms] T_{sleep}: Sleep period [ms] T_0: Discovery interval [s] T_{cw}: Contention window size [ms] T_{hdr}: Control packets (RTS/CTS/Ack) duration [ms]	T_{slot}^* $T_{guard} + T_{cw} + T_{hdr}$ $[0.05, 0.5] * T_{slot}$ $T_{slot} - (T_{active} + T_{sync})$ 360 $15 * 0.62$ $\frac{8+L_{pbl}}{R}$
DMAC	T_{frame}: Frame duration [ms] T_{sync}: Synchronization period [ms] N_{slots}: Number of slots per frame [#slots] T_{cw}: Contention window size [ms] T_{hdr}, T_{ack}: Pkt header & Ack duration [ms]	T_{frame}^* T_{sync}^* 32 15×0.62 $\frac{10+L_{pbl}}{R}$
LMAC	T_{frame}: Frame duration [ms] N_{slots}: Number of slots per frame [#slots] L_{data}^{max}: Maximum data length [byte] T_{data}^{max}: Maximum data length duration [ms] T_{hdr}: Packet header duration [ms]	T_{frame}^* $N_{slots} = \frac{T_{frame}}{T_{slot}}$ $32, 64, 128, 256$ $\frac{L_{data}^{max}}{R}$ $\frac{7+L_{pbl}}{R}$

nodes are therefore synchronized every $(C+1)$ slots and require a clock-drift compensation of $T_{guard} = 2\theta T_{slot}(C + 1)$. The synchronization phase has a length of $T_{sync} = T_{guard} + T_{cw} + T_{hdr}$. In the active phase, the nodes contend for the channel using RTS/CTS handshake with acknowledgment to send data packets (6). By default, nodes perform carrier sensing in active mode and switch off their radios for the duration indicated in the received RTS/CTS packets to avoid overhearing. It can be observed that the T_{active} and T_{sleep} are the key SMAC's parameters tunable to optimize the energy and delay performance since by knowing these two parameters, T_{slot} and T_{sync} can be calculated. Hence, the vector parameters for SMAC protocol is given by $X_{SMAC} = [T_{active}, T_{sleep}]$. In SMAC, energy is spent in the synchronization and in active phases, except when background traffic is exchanged [Langendoen and Meier 2010]. Thus, the per-node energy consumption based on the protocol operation modes, the e2e packet delay, and the bottleneck constraint are given in the following equations: (description of every term used in the formulas can be found in table I and table VI):
a) The Energy of node n:

$$E^n = E_{\text{active}}^n + E_{\text{sync}}^n,\tag{22}$$

where

$$E_{\text{active}}^n = \frac{T_{up} + T_{\text{active}}}{T_{\text{slot}}} - (T_{\text{data}} - (T_{up} + T_{\text{hdr}}))\, F_B^n,$$

$$E_{\text{sync}}^n = \frac{T_{\text{sync}}}{T_{\text{slot}}} + \frac{T_{\text{sleep}}}{T_0},$$

b) The delay of node n at level d:
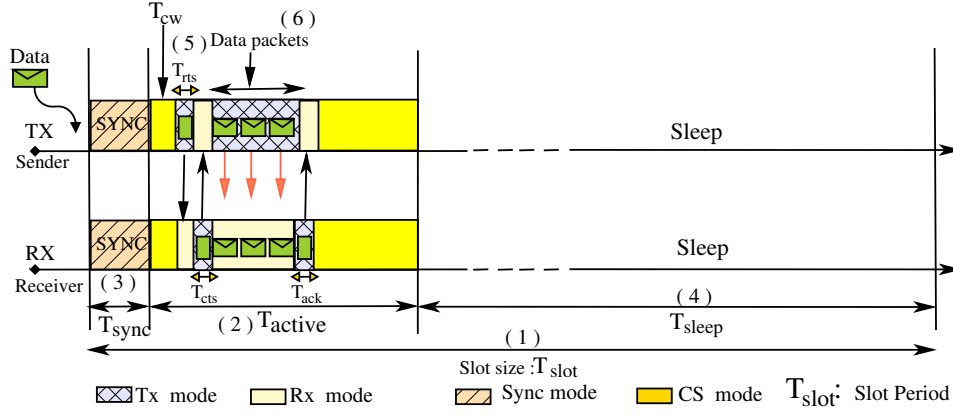
Fig. 23.   SMAC's synchronization, active, and sleep modes.

$$L^n = T_{\text{init}} + \sum_{i=1}^{d} \left(\frac{i}{H_{\text{active}}} T_{\text{slot}} + i\, H_{\text{active}} \left(\frac{T_{cw}}{2} + T_{\text{data}} \right) \right), \tag{23}$$

where $T_{\text{init}} = (T_{\text{sleep}} + T_{\text{sync}})/2$, $H_{\text{active}} = \lceil \frac{T_{\text{active}}}{\frac{T_{cw}}{2} + T_{\text{data}}} \rceil$, and $T_{\text{data}} = 4T_{\text{hdr}} + P/R$.

c) The bottleneck constraints:

$$(T_{cw}/2 + T_{\text{data}})F_I^0 < T_{\text{active}}/T_{\text{slot}}/4, \tag{24}$$
$$T_{\text{active}} \geqslant T_{cw} + T_{\text{data}}. \tag{25}$$

From equations (22-24), the following system wide energy-delay functions are derived:

a) The network energy consumption function:

$$\boldsymbol{E}^{\text{SMAC}} = \max_{n \in N} \left(\frac{\lambda_1}{T_{slot}} + \frac{T_{active}}{T_{slot}} + \frac{\lambda_2}{T_{sleep}} + \lambda_3 \right) \tag{26}$$

where $\lambda_1 = \sum_{n=1}^{N}(T_{cw} + T_{hdr})$, $\lambda_2 = \sum_{n=1}^{N} \frac{1}{T_\theta}$, and $\lambda_3 = (2\theta - (T_{data} - T_{hdr} - T_{up})F_B^n)$.

b) The e2e packet delay function:

$$\boldsymbol{L}^{\text{SMAC}} = \max_{n \in N} \left(\mu_1 T_{slot} + \mu_2 T_{sleep} + \frac{\mu_3 T_{slot}}{T_{active}} + \frac{\mu_4}{T_{active}} + \mu_5 \right) \tag{27}$$

where $\mu_1 = \theta$, $\mu_2 = 1/2$, $\mu_3 = \sum_{i=1}^{d}(\frac{T_{cw}}{2} + T_{data})$, $\mu_4 = \sum_{i=1}^{d}(\frac{T_{cw}}{2} + T_{data})^2$, and $\mu_5 = \frac{T_{cw} + T_{hdr}}{2}$.

*A.1.3. **Frame-based***. **DMAC** (Data-gathering MAC) by [Lu et al. 2007] is a synchronous protocol that address the delay for convergecast based data delivery, by scheduling the receiving and sending slots in a wave-like chain according to the nodes level in the gathering tree. Each frame, T_{frame}, is composed by a reception slot, a transmission slot, and a number of sleep slots (1), Fig 24.b). The reception slot of each node coincides with the transmission slot of down-level nodes (children) (2). Whereas, the transmission slot coincides with the reception slot of the parent (3).

Fig. 24. DMAC's transmission, receiving, and data prediction modes.

Nodes must contend in each slot T_{slot} using CSMA with acknowledgments (4) and compensate with a guard time $T_{\text{guard}} = 4\theta T_{\text{sync}}$ for clock drift (5). Synchronization messages are exchanged every T_{sync} using regular transmission and reception slots (6). Additional slots are added to the sleep period when there is more data to be sent. This is using a data-prediction scheme (7). Similarly to X-MAC, the main adjustable parameters that impact the energy and delay performance are the frame size T_{frame} and the synchronization period T_{sync}. Thus, the tunable DMAC parameters are given by $X_{\text{DMAC}} = [T_{\text{frame}}, T_{\text{sync}}]$. Bellow, the per-node energy, e2e packet delay and bottleneck constraint of DMAC protocol are detailed. In DMAC, the synchronization frequency F_{sync} is $1/T_{\text{sync}}$, if the output traffic frequency F_{out} is lower than the synchronization period, and it is equal to 0 otherwise. Due to low data rate application assumption, we derive the energy equations only for the former case i.e. $F_{\text{sync}} = 1/T_{\text{sync}}$.

a) The energy of node n:

$$E^n = E_{rx}^n + E_{tx}^n + E_{dp}^n \tag{28}$$

where $E_{rx}^n = \frac{T_{up} + 4\theta T_{\text{sync}}/2 + T_{cw} + T_{\text{data}}}{T_{\text{frame}}} + \frac{T_{cs} + T_{\text{hdr}}}{T_{\text{sync}}}$, $E_{tx}^n = (T_{cs} + T_{\text{data}}) F_{\text{out}}^n$ and $E_{dp}^n = (F_I^n + |I_d| F_{\text{sync}}) (T_{up} + 2\theta T_{\text{sync}} + T_{cw} + T_{\text{data}})$.

b) The delay of node n at level d:

$$L^n = \frac{T_{\text{frame}}}{2} + \sum_{i=1}^{d} \left(\frac{4\theta T_{\text{sync}}}{2} + T_{cw} + T_{\text{data}} \right) \tag{29}$$

where $T_{data} = T_{hdr} + P/R + T_{ack}$.

c) The bottleneck constraint:

$$(F_I^0 + |I^0| F_{\text{sync}}^1) T_{\text{frame}} < 1/2. \tag{30}$$

From equations (28-30), the following system wide energy-delay functions are derived:

a) The network energy consumption function:

$$\boldsymbol{E}^{\text{DMAC}} = \max_{n \in N} \left(\frac{\gamma_1}{T_{\text{frame}}} + \gamma_2 \frac{T_{\text{sync}}}{T_{\text{frame}}} + \gamma_3 T_{\text{sync}} + \frac{\gamma_4}{T_{\text{sync}}} + \gamma_5 \right) \tag{31}$$

where $\gamma_1 = T_{up} + T_{cw} + T_{\text{data}}$, $\gamma_2 = 2\theta$, $\gamma_3 = 2\theta$, $\gamma_4 = T_{cs} + T_{\text{hdr}} + (T_{up} + T_{cw} + T_{\text{data}}) I^n$ and $\gamma_5 = (T_{cs} + T_{\text{data}}) F_{\text{out}}^n + (T_{up} + T_{cw} + T_{\text{data}}) F_I^n + 2\theta I^n$.
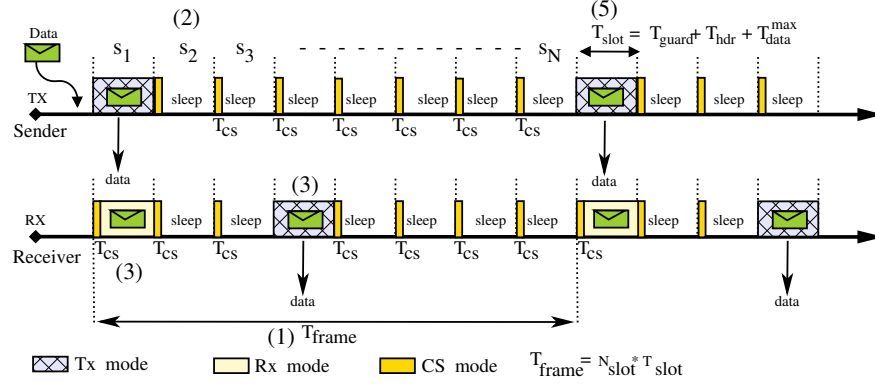
Fig. 25.   LMAC's carrier sense, transmission, and receiving modes.

b) The e2e packet delay function:

$$L^{\text{DMAC}} = \max_{n \in N} \left(\delta_1 \, T_{\text{frame}} + \delta_2 \, T_{\text{sync}} + \delta_3 \right) \tag{32}$$

where $\delta_1 = 1/2$, $\delta_2 = \sum_{i=1}^{d} 2\theta$ and $\delta_3 = \sum_{i=1}^{d} (T_{cw} + T_{\text{data}})$.

LMAC (Lightweight MAC) by [van Hoesel and Havinga 2004] is a self-organizing TDMA scheme that organizes time into frames (1), Fig. 25. It relies on a distributed collision-free slot selection mechanism in the 2-hop neighborhood. Each frame, T_{frame}, is composed of $N_{\text{slots}} = s_1, s_2, .., s_N$ (2). Every node owns one transmission slot in which it sends its data packet without acknowledgment (3); the recovery from external interference is left to the upper layers. A node must listen (i.e., perform a carrier sense T_{cs}) in all slots other than its own to check for incoming data (4). Synchronization is performed using packet headers in every occupied slot. Therefore, the slot owner has to guard for the maximum clock drift. The slot must fit the guard-time, the packet header, and the maximum data payload $T_{\text{data}}^{\text{max}}$ (5) where, $T_{\text{slot}} = T_{\text{guard}} + T_{\text{hdr}} + T_{\text{data}}^{\text{max}}$ and $T_{\text{frame}} = N_{\text{slots}} T_{\text{slot}}$. Thus, changing the frame size will change the slot size and the number of slots accordingly. Following the LMAC operation model, the main adjustable parameter that impacts the energy and delay performance is the frame size, T_{frame}. Therefore, the tunable LMAC parameter is given by $X_{\text{LMAC}} = [T_{\text{frame}}]$. The LMAC per-node energy, e2e packet delay, and bottleneck constraint are given in the following:

a) The energy of node n:

$$E^n = E^n_{cs} + E^n_{\text{hdr}} + E^n_{tx} + E^n_{rx}, \tag{33}$$

where $E^n_{cs} = (N_{\text{slots}} - 1) \frac{T_{cs}}{T_{\text{frame}}}$, $E^n_{\text{hdr}} = C \frac{T_{\text{guard}}/2 + T_{\text{hdr}}}{T_{\text{frame}}}$, $E^n_{tx} = \frac{T_{up} + T_{\text{guard}} + T_{\text{hdr}}}{T_{\text{frame}}} + F^n_{\text{out}} (P/R)$ and $E^n_{rx} = F^n_I (P/R)$.

b) The delay of node n at level d:

$$L^n = \frac{T_{\text{frame}} + T_{\text{slot}}}{2} + \sum_{i=1}^{d-1} \left((N_{\text{slots}} - 1) \frac{T_{\text{slot}}}{2} \right) - (T_{\text{data}}^{\text{max}} - T_{\text{data}}), \tag{34}$$

where $T_{\text{data}} = P/R$.

c) The bottleneck constraint:

$$F_{out}^1 \, T_{frame} < 1/2. \tag{35}$$

From equations (33-35), the following system wide energy-delay functions are derived:

a) The network energy consumption function:

$$\boldsymbol{E}^{\text{LMAC}} = \max_{n \in N} \left(\frac{\lambda_1}{\lambda_2 T_{frame} + \lambda_3} + \frac{\lambda_4}{T_{frame}} + \lambda_5 \right) \tag{36}$$

where $\lambda_1 = T_{cs}$, $\lambda_2 = 4\theta$, $\lambda_3 = T_{hdr} + T_{data}^{max}$, $\lambda_4 = T_{up} + T_{hdr} + C\,T_{hdr} - 1$ and $\lambda_5 = (F_{out}^n + F_I^n)\,P/R + 2\theta(C+2)$.

b) The e2e packet delay function:

$$\boldsymbol{L}^{\text{LMAC}} = \max_{n \in N} \left(\zeta_1 T_{frame} + \zeta_2 \right) \tag{37}$$

where $\zeta_1 = \sum_{i=1}^{d} 1/2 + \sum_{i=1}^{d-2} 2\theta$ and $\zeta_2 = \sum_{i=1}^{d-2} \left(\frac{T_{hdr} + T_{data}^{max}}{2} \right) - (T_{data}^{max} - T_{data})$.

A.2. Framework Application

*A.2.1. **Energy Optimization**.* The energy optimization subject to maximum e2e packet delay given the application requirements in terms of e2e packet delay bound $L_{\max}$, derives optimal MAC parameters for B-MAC, X-MAC, SMAC, DMAC, and LMAC protocols:

$$
\begin{array}{ll}
(\boldsymbol{P5}) & \min \; E^{\text{BMAC}}(T_w) \\
& \text{s.t.} \;\; L^{\text{BMAC}}(T_w) \leqslant L_{\max} \\
& \quad\;\; T_w \geqslant T_w^{\min} \\
& \quad\;\; |I^0|\, E_{tx}^1 \leqslant 1/4 \\
& \text{var.} \; T_w
\end{array}
\qquad
\begin{array}{ll}
(\boldsymbol{P6}) & \min \; E^{\text{XMAC}}(T_w) \\
& \text{s.t.} \;\; L^{\text{XMAC}}(T_w) \leqslant L_{\max} \\
& \quad\;\; T_w \geqslant T_w^{\min} \\
& \quad\;\; |I^0|\, E_{tx}^1 \leqslant 1/4 \\
& \text{var.} \; T_w
\end{array}
$$

$$
\begin{array}{ll}
(\boldsymbol{P7}) & \min \; E^{\text{SMAC}}(T_{\text{active}}, T_{\text{sleep}}) \\
& \text{s.t.} \;\; L^{\text{SMAC}}(T_{\text{active}}, T_{\text{sleep}}) \leqslant L_{\max} \\
& \quad\;\; T_{\text{active}} + T_{\text{sleep}} \geqslant T_{\text{slot}}^{\min} \\
& \quad\;\; (T_{cw}/2 + T_{data})F_I^0 < T_{\text{active}}/(4T_{\text{slot}}) \\
& \quad\;\; T_{\text{active}} \geqslant T_{cw} + T_{data} \\
& \text{var.} \; T_{\text{active}}, T_{\text{sleep}}
\end{array}
$$

$$
\begin{array}{ll}
(\boldsymbol{P8}) & \min \; E^{\text{DMAC}}(T_{\text{frame}}, T_{\text{sync}}) \\
& \text{s.t.} \;\; L^{\text{DMAC}}(T_{\text{frame}}, T_{\text{sync}}) \leqslant L_{\max} \\
& \quad\;\; T_{\text{frame}} \geqslant T_{\text{frame}}^{\min} \\
& \quad\;\; (F_I^0 + |I^0|F_{\text{sync}})\,T_{\text{frame}} < 1/2 \\
& \quad\;\; F_{out}^n \leqslant \frac{1}{T_{\text{sync}}} \\
& \text{var.} \; T_{\text{frame}}, T_{\text{sync}}
\end{array}
\qquad
\begin{array}{ll}
(\boldsymbol{P9}) & \min \; E^{\text{LMAC}}(T_{\text{frame}}) \\
& \text{s.t.} \;\; L^{\text{LMAC}}(T_{\text{frame}}) \leqslant L_{\max} \\
& \quad\;\; T_{\text{frame}} \geqslant T_{\text{frame}}^{\min} \\
& \quad\;\; F_{out}^1 \, T_{\text{frame}} < 1/2 \\
& \text{var.} \; T_{\text{frame}}
\end{array}
$$

*A.2.2. **Delay Optimization**.* The delay optimization subject to energy consumption given the application requirements in terms of initial energy budget E_{budget} provides the optimal MAC parameters for the MAC protocols:

$$
\begin{array}{ll}
(\boldsymbol{P5'}) & \min \; L^{\text{BMAC}}(T_w) \\
& \text{s.t.} \;\; E^{\text{BMAC}}(T_w) \leqslant E_{\text{budget}} \\
& \quad\;\; T_w \geqslant T_w^{\min} \\
& \quad\;\; |I^0|\, E_{tx}^1 \leqslant 1/4 \\
& \text{var.} \; T_w
\end{array}
\qquad
\begin{array}{ll}
(\boldsymbol{P6'}) & \min \; L^{\text{XMAC}}(T_w) \\
& \text{s.t.} \;\; E^{\text{XMAC}}(T_w) \leqslant E_{\text{budget}} \\
& \quad\;\; T_w \geqslant T_w^{\min} \\
& \quad\;\; |I^0|\, E_{tx}^1 \leqslant 1/4 \\
& \text{var.} \; T_w
\end{array}
$$

$(\mathbf{P7'})$ $\min$ $L^{\text{SMAC}}(T_{\text{active}}, T_{\text{sleep}})$

   s.t. $E^{\text{SMAC}}(T_{\text{active}}, T_{\text{sleep}}) \leqslant E_{\text{budget}}$

       $T_{\text{active}} + T_{\text{sleep}} \geqslant T_{\text{slot}}^{\min}$

       $(T_{cw}/2 + T_{\text{data}})F_I^0 < T_{\text{active}}/(4T_{\text{slot}})$

       $T_{\text{active}} \geqslant T_{cw} + T_{\text{data}}$

   var. $T_{\text{active}}, T_{\text{sleep}}$

$(\mathbf{P8'})$ $\min$ $L^{\text{DMAC}}(T_{\text{frame}}, T_{\text{sync}})$

   s.t. $E^{\text{DMAC}}(T_{\text{frame}}, T_{\text{sync}}) \leqslant E_{\text{budget}}$

       $T_{\text{frame}} \geqslant T_{\text{frame}}^{\min}$

       $(F_I^0 + |I^0|F_{\text{sync}})\,T_{\text{frame}} < 1/2$

       $F_{\text{out}}^n \leqslant \frac{1}{T_{\text{sync}}}$

   var. $T_{\text{frame}}, T_{\text{sync}}$

$(\mathbf{P9'})$ $\min$ $L^{\text{LMAC}}(T_{\text{frame}})$

   s.t. $E^{\text{LMAC}}(T_{\text{frame}}) \leqslant E_{\text{budget}}$

       $T_{\text{frame}} \geqslant T_{\text{frame}}^{\min}$

       $F_{\text{out}}^1 T_{\text{frame}} < 1/2$

   var. T_{frame}

A.2.3. ***Energy-Delay Trade-off***. The equivalent concave problems of B-MAC, X-MAC, SMAC, DMAC, and LMAC protocols after game theory framework application are expressed by:

$(\mathbf{P5^*})$ $\max$ $log(E_{\text{worst}}^{\text{BMAC}} - E_1) + log(L_{\text{worst}}^{\text{BMAC}} - L_1)$

   s.t. $E_{\text{worst}}^{\text{BMAC}} \geqslant E^{\text{BMAC}}(T_w)$

       $E_1 \geqslant E^{\text{BMAC}}(T_w)$

       $L_{\text{worst}}^{\text{BMAC}} \geqslant L^{\text{BMAC}}(T_w)$

       $L_1 \geqslant L^{\text{BMAC}}(T_w)$

       $T_w \geqslant T_w^{\min}$

       $|I^0|\,E_{tx}^1 \leqslant 1/4$

   var. E_1, L_1, T_w

$(\mathbf{P6^*})$ $\max$ $log(E_{\text{worst}}^{\text{XMAC}} - E_1) + log(L_{\text{worst}}^{\text{XMAC}} - L_1)$

   s.t. $E_{\text{worst}}^{\text{XMAC}} \geqslant E^{\text{XMAC}}(T_w)$

       $E_1 \geqslant E^{\text{XMAC}}(T_w)$

       $L_{\text{worst}}^{\text{XMAC}} \geqslant L^{\text{XMAC}}(T_w)$

       $L_1 \geqslant L^{\text{XMAC}}(T_w)$

       $T_w \geqslant T_w^{\min}$

       $|I^0|\,E_{tx}^1 \leqslant 1/4$

   var. E_1, L_1, T_w

$(\mathbf{P7^*})$ $\max$ $log(E_{\text{worst}}^{\text{SMAC}} - E_1) + log(L_{\text{worst}}^{\text{SMAC}} - L_1)$

   s.t. $E_{\text{worst}}^{\text{SMAC}} \geqslant E^{\text{SMAC}}(T_{\text{active}}, T_{\text{sleep}})$

       $E_1 \geqslant E^{\text{SMAC}}(T_{\text{active}}, T_{\text{sleep}})$

       $L_{\text{worst}}^{\text{SMAC}} \geqslant L^{\text{SMAC}}(T_{\text{active}}, T_{\text{sleep}})$

       $L_1 \geqslant L^{\text{SMAC}}(T_{\text{active}}, T_{\text{sleep}})$

       $T_{\text{active}} + T_{\text{sleep}} \geqslant T_{\text{slot}}^{\min}$

       $(T_{cw}/2 + T_{\text{data}})F_I^0 < T_{\text{active}}/(4T_{\text{slot}})$

       $T_{\text{active}} \geqslant T_{cw} + T_{\text{data}}$

   var. $E_1, L_1, T_{\text{active}}, T_{\text{sleep}}$

$(\mathbf{P8^*})$ $\max$ $log(E_{\text{worst}}^{\text{DMAC}} - E_1) + log(L_{\text{worst}}^{\text{DMAC}} - L_1)$

   s.t. $E_{\text{worst}}^{\text{DMAC}} \geqslant E^{\text{DMAC}}(T_{\text{frame}}, T_{\text{sync}})$

       $E_1 \geqslant E^{\text{DMAC}}(T_{\text{frame}}, T_{\text{sync}})$

       $L_{\text{worst}}^{\text{DMAC}} \geqslant L^{\text{DMAC}}(T_{\text{frame}}, T_{\text{sync}})$

       $L_1 \geqslant L^{\text{DMAC}}(T_{\text{frame}}, T_{\text{sync}})$

       $T_{\text{frame}} \geqslant T_{\text{frame}}^{\min}$

       $(F_I^0 + |I^0|F_{\text{sync}})\,T_{\text{frame}} < 1/2$

       $F_{\text{out}}^n \leq \frac{1}{T_{\text{sync}}}$

   var. $E_1, L_1, T_{\text{frame}}, T_{\text{sync}}$

$(\mathbf{P9^*})$ $\max$ $log(E_{\text{worst}}^{\text{LMAC}} - E_1) + log(L_{\text{worst}}^{\text{LMAC}} - L_1)$

   s.t. $E_{\text{worst}}^{\text{LMAC}} \geqslant E^{\text{LMAC}}(T_{\text{frame}})$

       $E_1 \geqslant E^{\text{LMAC}}(T_{\text{frame}})$

       $L_{\text{worst}}^{\text{LMAC}} \geqslant L^{\text{LMAC}}(T_{\text{frame}})$

       $L_1 \geqslant L^{\text{LMAC}}(T_{\text{frame}})$

       $T_{\text{frame}} \geqslant T_{\text{frame}}^{\min}$

       $F_{\text{out}}^1 T_{\text{frame}} < 1/2$

   var. $E_1, L_1, T_{\text{frame}}$

ELECTRONIC APPENDIX

The electronic appendix for this article can be accessed in the ACM Digital Library.