A Distributed Protocol for Cooperation Among Different Wireless Sensor Networks

Pedro O.S. Vaz de Melo*, Felipe D. Cunha*, Antonio A.F. Loureiro*

*Federal University of Minas Gerais, Belo Horizonte, Brazil,
Email: olmo, fdcunha, loureiro@dcc.ufmg.br

***Abstract*—An important issue in the design of a wireless sensor network (WSN) is to devise techniques to make efficient use of its energy, and thus, extend its lifetime. When two or more WSNs are deployed in the same place and their sensors cooperate with the other networks, they may improve their operability, by extending its lifetime by trading routing favors or increasing the data entropy by a common data aggregation. Despite being obvious and simple, this idea brings with it many implications that hinder cooperation between the networks. Whereas a WSN has a rational and selfish character, it will only cooperate with another WSN if this provides services that justify the cooperation. The goal of this work is to present the Virtual Cooperation Bond (VCB) protocol, that is a distributed protocol that makes different WSNs to cooperate, enabling cooperation if, and only if, all the different WSNs benefit with the cooperation. In the simulation results, we consider WSNs with different configurations and we show that the proposed protocol enables cooperation solely when the cooperation is beneficial to both networks, and in this case, it saves their energies and extends their lifetimes.**

I. Introduction

Wireless Sensor Networks (WSNs) [1] may be deployed on the most varied places, like forests, volcanoes and deserts. Furthermore, a possibly forthcoming scenario will be distinct WSNs owned by different authorities working at the same place, serving as a supporting tool for a wide variety of applications. In the Amazon forest, for instance, we could have a WSN, owned by the government, deployed for detecting fires and another WSN, owned by a non-governmental organization, deployed for detecting the movement of specific species of animals. Moreover, both size and production cost of a sensor node should be as small as possible and its battery replacement will be rarely possible. As a consequence, a sensor has severe constraints in terms of processing, memory, communication and energy capacity. In particular, depending on the sensing application, data communication can consume a significant amount of energy [2]. Thus, to assure an efficient operability of the WSN, sensor nodes should cooperate with each other to improve the quality of the data and to reduce the energy consumption in the communication process.

In this direction, when two WSNs, installed at the same place, share their sensor nodes in the execution of one or more activities in a profitable way, both networks may improve their capabilities and perform their activities in a more efficient way. Despite being obvious and simple, this idea brings several implications that hinder cooperation among the networks. We have to assume that a WSN has a rational and selfish character and will only cooperate with another sensor network if this association provides services that justify the cooperation, as illustrated in Fig. 1.

An interesting technique to model conflict situations among two or more rational and selfish agents is through the concept of game theory [3], [4]. Generally speaking, game theory is based on models that express the interaction among players by modeling them as rational and selfish agents in such a way that they act to maximize their own utility. This allows the analysis of existing algorithms and protocols for WSNs, as well as the design of equilibrium-inducing mechanisms that provide incentives for individual nodes to behave in socially-constructive ways. Game theory defines mechanisms for players to choose the best available action, but at the same time it provides a scenario where other players' utilities can also be maximized. By modeling the problem of cooperation among different WSNs as games, the networks' behaviors and actions can be analyzed in a formalized game structure, by which the theoretical achievements in game theory can be fully utilized. Each authority that governs its network wishes to both maximize its lifetime and its quality of service, and will only cooperate with another network if this cooperation brings benefits.

In this work we present a game-theoretic formulation for the problem of cooperation among different WSNs deployed in the same location. We show that the only way to achieve cooperation is through a protocol that enables a joint strategy change by the nodes. Moreover, since WSNs may scale up to hundreds of thousands of nodes, cooperation should be established and maintained in a distributed fashion. Therefore, in this work we propose the *Virtual Cooperation Bond* (VCB) protocol that is able to make the networks cooperate if and only if the cooperation is beneficial to them. VCB creates autonomous 4-node cooperation structures distributed along the deployment environment, and each structure is maintained solely by the four nodes that comprise it.

The rest of this paper is organized as follows. We describe the related work in Section II. The game-theoretic formulation is presented in Section III. In Section IV, we describe the proposed VCB protocol and, in Section V, the simulation results. Finally, we conclude this paper in Section VI.

II. Related Work

Felegyhazi et al. [5] addressed the problem of cooperation among different WSNs. The strategies of the owners of the net-

978-1-4673-3122-7/13/$31.00 ©2013 IEEE

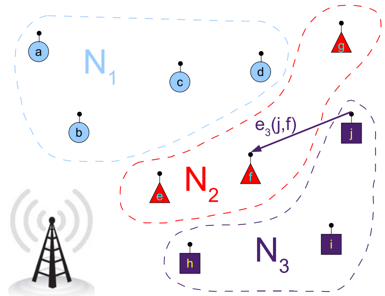


Fig. 1. A scenario where different WSNs are deployed in the same location. We see three networks, N_1, N_2 and N_3, and a possible cooperation edge $e_3(j, f)$, where node f accepts data of network N_3 coming from sensor node j.

works are set if their nodes forward messages coming from other networks and if they ask other networks to forward their messages. It is assumed that sensors send messages periodically and synchronously to their respective sinks and these, in turn, send to their nodes a bit telling if the data collection was satisfactory. From this, nodes control their strategies to minimize their energy consumption and maximize the data collection rate. In this model, the networks converge essentially to two equilibria, a non-cooperative in that no node provides and asks for services to another network, and a cooperative, which all nodes provide and ask for services of another network.

As showed in [6], [7], the problem of cooperation among different WSNs involves several parameters that can significantly influence the establishment of cooperation and its benefits. A solution that enables cooperation only when all of the sensor nodes cooperate is not practical, since it is very unlikely that two different WSNs have the same configuration homogeneously through out the deployment area. An efficient practical solution should be distributed, allowing parts of the networks to cooperate where others do not.

Another crucial factor to establish cooperation is the knowledge of the costs and the benefits that will incur to the networks by cooperating. Wu and Shu [8] showed that if sensors declare their cost to route messages, the networks are naturally encouraged to cooperate and to share their nodes. Moreover, they showed that if the agent reveals its real cost for routing a packet, its benefits will be maximized.

Finally, Crosby and Pissinou [9] demonstrated that cooperation is not evolutionary stable when the networks have mobile sensor nodes and are playing the iterated N-player prisoner's dilemma. However, they showed that in the case of stationary sensor nodes, there is some possibility for cooperation to emerge without any incentive when the same game is played. Our work goes in this direction and, in the next section, we model the problem of cooperation among different WSNs as the iterated N-player prisoner's dilemma and we propose a distributed strategy to establish cooperation among the networks.

III. WSN Cooperation Game

In this section, we propose a new formulation for the problem of cooperation among different WSNs deployed at the same location. It is similar to the ones discussed earlier, but we aim to guarantee cooperation in a distributed and ad-hoc fashion, for more realistic and dynamic scenarios. Let N be a set of m networks, $N_1, ..., N_m$, and let n_0 be the sink node, which is shared by all networks. Each network N_i, $1 \leq i \leq m$, has a set of unique sensor nodes $N_i = \{n_i^1, n_i^2, ..., n_i^{|N_i|}\}$ and the sink node n_0 is shared by all networks. We model the cooperation among the networks as a multigraph $G = (N, E)$, where N is the set of all networks and E is the set of edges between an arbitrary pair of sensor nodes. For the sake of simplicity, we model the traffic originating at each network in a unique way. Thus, there will be a distinct edge for the collected data packets coming from each network and sent towards the sink. All edges that carry traffic from network N_i are represented by $e_i(n_j, n_k)$, where j and k are not necessarily different and $1 \leq j, k \leq m$. The set of edges e_i represents the routing infrastructure responsible for carrying the traffic from network N_i. From a practical point of view, a communication link between a pair of nodes may represent a set of distinct edges.

For simplicity, we consider that the initial data collecting routing structure of each network is a routing spanning tree [10], but any routing structure can be used. We consider that the routing spanning tree is the best tree that each network is able to generate following a defined QoS parameter, being optimal or not. Thus, each node n_i^u has, at most, a single outgoing edge $e_i(n_i^u, n_j^v)$ leaving n_i^u for each N_j, i.e., n_i^u can forward messages from N_j to at most one node. The set of edges that leaves n_i^u is denoted by $E_i^u, |E_i^u| \leq m$ and the weight $w(e_i(n_i^u, n_j^v)) > 0$ of edge $e_i(n_i^u, n_j^v)$ is independent of N_i, being given by the energy consumption model $w(e_i(n_i^u, n_j^v)) = f_{rx} + f_{tx} \times d(n_i^u, n_j^v)^\alpha$, where f_{rx} is the cost to receive a message, f_{tx} is the cost to send a message to a unit of distance d, $d(n_i^u, n_j^v)$ is the distance between n_i^u and n_j^v and α is the path loss exponent. The energy model w may be changed without invalidating this formulation.

The players in our game-theoretic formulation are the m networks and the strategy of N_i is to set $S_i = \{E_i^u \ \forall \ n_i^u \in N_i\}$, i.e., set of nodes N_i will relay data collection messages. The cost in energy c_i^u that each node n_i^u incurs for its network is given by $c_i^u = \sum_{i=1}^{m} w(E_i^u)$, sum of weights of edges that leave node n_i^u. The utility function that maps the payoff Π_i of each network N_i, given its topology, is $\Pi_i = \sum_{\forall n_i^u \in N_i} (\phi - c_i^u)$, where ϕ is an abstract value of the data that is transmitted, where $\phi \gg c_i^u \ \forall \ n_i^u \in \ N_i$. It is natural that the value ϕ of the data be significantly higher than the cost to transmit it, since a WSN only exists to collect data, thus, without data there is no network. The payoff Π_i captures the data collection rate and the energy consumption in routing of network N_i, being determinant to define if the network cooperates or not with another network.

Initially, no node n_i^u is cooperating with other networks besides its own network N_i. N_i, which is a rational and selfish player, should change its strategy and make one or more of its nodes to forward messages coming from other networks iff it increases its payoff Π_i. However, if all networks maintain their strategies and N_i changes its original strategy and makes a node n_i^u to cooperate and forward messages from other networks, its cost c_i^u will increase, making the payoff Π_i to decrease. Thus, initially, the game is in Nash equilibrium because it is not possible for a network to increase its payoff changing its strategy if all other networks maintain theirs. This makes N_i to only be able to increase its payoff if it coordinates with another network N_j a joint change of strategies, $S_i \rightarrow S_i', S_j \rightarrow S_j'$, so that S_i' benefits N_j and S_j' benefits N_i. In game theory, a game that allows two or more players to coordinate their actions is called a cooperative game [3].

We can model this scenario as a cooperative, non-zero sum, two-player game, where the game is played repeatedly during the lifetimes of N_i and N_j. If both networks cooperate, their payoffs will be a "reward" R_i for N_i and R_j for R_j, that are greater than their non cooperative payoffs. This game (WSN-Cooperation game) is similar to the Iterated Prisoner's Dilemma showed in Fig. 2, having the "reward" payoff R, the "punishment" payoff P, the "temptation" payoff T and the "sucker's payoff" S [11]. In the ordered pairs we see the outcomes of the game given the strategies of each player, networks N_i and N_j, with the first payoff given to the row player, N_i, and the second given to the column player, N_j. Given the inequality $T > R > P > S$, not to cooperate is always the best strategy, resulting in the worst outcome when both networks play it. However, as in the Iterated Prisoner's Dilemma game, if we have the inequalities $R_i + R_j > T_i + S_j$ and $R_i + R_j > T_j + S_i$, then mutual cooperation returns the highest collective payoff and we can coordinate networks to jointly cooperate and get better payoffs. Moreover, if we guarantee that if one defects, it will be punished by the other by breaking the cooperation agreement. The WSN-Cooperation game has a cooperation stable solution by the Folk's Theorem [3].

	Cooperate	Not Cooperate
Cooperate	P_i, P_j	S_i, T_j
Not Cooperate	T_i, S_j	R_i, R_j

Fig. 2. Payoff matrix of the "WSN-Cooperation" game.

IV. The Virtual Cooperation Bond

A. A Solution to the Game

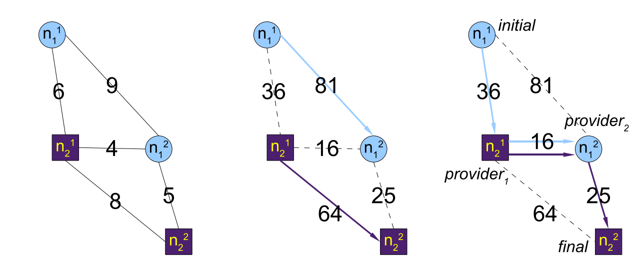
A possible joint strategy of two networks N_i and N_j is each one to create a cooperation edge that harms itself but also benefits the other network, with benefits greater than losses. Formally, a node n_i^u from N_i creates an outgoing edge $e_j(n_i^u, n_j^v)$ to a node n_j^v from network N_j, meaning it can relay messages from N_j destined to node n_j^q using edge $e_j(n_i^u, n_j^v)$. Similarly, the same process occurs in N_j with a node n_j^p creating an outgoing edge $e_i(n_j^p, n_i^q)$ to n_i^q. If both networks N_i and N_j use these cooperation edges $e_j(n_i^u, n_j^v)$ and $e_i(n_j^p, n_i^q)$ fairly, in a way that increase the benefits of using the external cooperation edge, then cooperation may be established and be stable.

The cooperation strategy that two edges $e_j(n_i^u, n_j^v)$ and $e_i(n_j^p, n_i^q)$ are created satisfying these restrictions will be called the Give to Conquer (GTC) strategy. As GTC is repeated to create different pairs of edges, the payoff of the networks involved in the process increases, since GTC guarantees that the benefits with the edge creation are greater than the losses. Thus, after creating all pairs of edges, the payoff of the networks is the maximum that can be achieved with GTC, being, a Pareto Optimal solution [3], since no network is able to increase its payoff without reducing the payoff of another.

The best way to make different WSNs to cooperate, according to the criteria of robustness and scalability, is making them interact with each other in a completely autonomous way. One must consider that they may change their topologies constantly, and new networks may be deployed around them and be part of the cooperation. Thus, network nodes should be responsible for executing the GTC strategy, which may locally control nodes asking for/providing favors. If a node cooperating observes in its local environment that its network is losing more than it is benefiting with the cooperation, via a *watchdog* [12] or other control policy, it can act immediately to stop cooperation in its neighborhood.

Fig. 3 illustrates a scenario that the GTC strategy may be executed in this way. Fig. 3-a depicts two nodes of N_1, n_1^1 and n_1^2, and two nodes of N_2, n_2^1 and n_2^2, as well as the distances among them. Fig. 3-b depicts the scenario with no cooperation, with node n_1^1 sending a message to n_1^2, node n_2^1 sending a message to n_2^2 and the transmission costs C_{tx}, whereas $C_{tx} = d^\alpha$ and $\alpha = 2$. The total transmission cost of network N_1 is 81, whereas of N_2 is 64. In this scenario, $E_1^1 = \{e_1 n_1^1, n_1^2\}$, $E_1^2 = \emptyset$, $E_2^1 = \{e_2 n_2^1, n_2^2\}$ and $E_2^2 = \emptyset$. The networks payoffs are $\Pi_1 = \phi - (81 f_{tx} + f_{rx})$ and $\Pi_2 = \phi - (64 f_{tx} + f_{rx})$.

In Fig. 3-c, a GTC strategy is executed and networks N_1 and N_2 start to cooperate. The GTC strategy creates two edges, $e_1 n_2^1, n_1^2 \in E_2^1$ and $e_2 n_1^2, n_2^2 \in E_1^2$, indicating that node $n_2^1 \in N_2$ cooperates with network N_1 and node $n_1^2 \in N_1$ cooperates with N_2. Then, node n_1^1, aware of the cooperation, removes edge $e_1 n_1^1, n_1^2$ and adds $e_1 n_1^1, n_2^1$ to E_1^1, since node n_2^1 is cooperating and will forward its messages. The same happens with node n_2^1, that removes edge $e_2 n_2^1, n_2^2$ and creates edge $e_2 n_2^1, n_1^2$. Established the cooperation, the payoffs of the networks are $\Pi_1 = \phi - (61 f_{tx} + 2 f_{rx})$ and $\Pi_2 = \phi - (32 f_{tx} + 2 f_{rx})$, which are greater than when they were not cooperating, whereas in this work $f_{tx} \gg f_{rx}$.



(a) Distance among the sensor nodes. (b) Transmissions costs without cooperation. (c) Transmissions costs with cooperation.

Fig. 3. The Virtual Cooperation Bond: a feasible model to execute the GTC strategy ($w(e) = d^2$).

The cooperation scenario shown in Fig. 3-c is called Virtual Cooperation Bond (VCB). In VCB, nodes of a network act as team players and work jointly to reduce the energy consumption of their networks. Nodes n_1^2 increased their cost to help their network. In the VCB strategy, involved nodes are neighbors and can observe their respective actions, i.e., the amount of favors their network requests and supplies, which becomes known to each node in the bond. Thus, the involved nodes control the credit/debit they have. This ensures the fairness in cooperation, i.e., no network provides favors at a higher rates that it receives, and vice versa, using only an upper limit Δ for the highest difference between favors provided/received.

B. Establishing the VCB

Before explaining the method that describes how the cooperation is set, we make some considerations. First, every single node has a unique identifier and knows its coordinates. Second, there is a pre-establish routing structure over the networks, in which every node has a target node that to send or forward its messages. Third, a sensor node is able to calculate the cost of a transmission, given the transmission distance.

Now we define roles to nodes in VCB as described in Fig. 3-c. The role *initial* is played by the relaying node n_1^1 of Fig. 3-c (the one that provides no favors), but benefits from the cooperation by relaying its messages to a node in a different network. The node that receives the message from *initial* is the one that has the role of *provider*$_1$, which is the node n_2^1 in Fig. 3-c, that besides providing favors to the node *initial*, also benefits from the cooperation by relaying its messages to the node that has the role of *provider*$_2$, which is from the other network and is represented by node n_1^2 in Fig. 3-c. Node *provider*$_2$ provides favors, and forwards messages from node *provider*$_1$ to node of role *final* that, in VCB, is only responsible for receiving these messages, and in Fig. 3-c, it is represented by node n_2^2.

In Fig. 4, we describe the process for obtaining a VCB. The first step is to broadcast through the networks the wish to cooperate. Each node that wants to cooperate sends a broadcast message of type `WantToCooperate` to all its neighbors with its current transmission cost. Thus, each node that wants to cooperate calculates locally if a VCB can be formed with its neighbors, guaranteeing that both networks will benefit with the cooperation. If so, it sends a message of type `AskCooperation` to the nodes in the VCB, informing what roles they should play in the VCB to be formed. If all nodes agree with the assignment roles and are not in another VCB, they send a message of type `AcceptCooperation` and cooperation between them is established. The node that sends the `AskCooperation` message is the node with the role *provider*$_1$, since its location in the VCB is the only one that has direct access to all the other nodes in the VCB.

The process described in Fig. 4 is executed only once for each VCB structure. Each node will send, at most, two setup (or overhead) messages (one `WantToCooperate` plus one `AskCooperation` or `AcceptCooperation`) to establish cooperation. The cooperation is maintained autonomously, but a *watchdog* [12] or other control policy is necessary to ensure that no node is *cheating*. The analysis of the possible overhead incurred by this control policy we leave for future work.

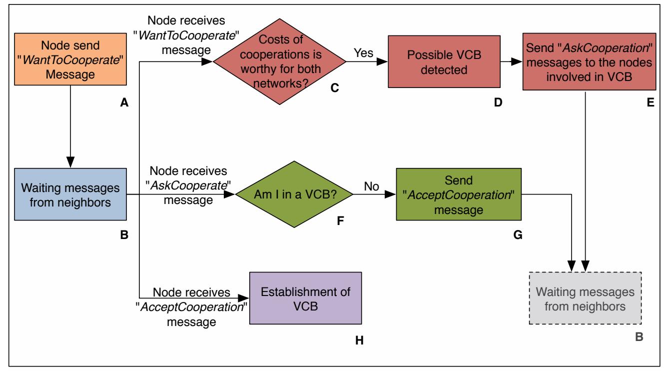


Fig. 4. The algorithm describing the establishment of the VCB.

V. Numerical Results

In this section, we present the simulation results of the proposed VCB protocol. All sensors from networks $N_1, N_2, ...$ are deployed in an area of $100 \times 100\, d^2$ with a flat topology. The communication range of each sensor starts with $30\, d$ and may be reduced depending on the routing structure. During $1000\,$s of simulation, events arrive in the network according to a Poisson Process with $\lambda_{Poisson}$ events/s. The default value of the path loss exponent is 4, and the number of nodes of each network is 50. The costs of communication are the same as the ones described in Section III. All simulation results correspond to the arithmetic mean of r simulations, in which r provides a good confidence to the results. For simplicity, we only show the results for network N_1 and we use as a baseline the scenario where no cooperation exists. We use the *Network Simulator 3.21* to make these simulations.

The first result shows the behavior of VCB when two networks (with same configuration) cooperate and the network density varies from 50 to 200 nodes. Fig. 5 shows the energy spent on transmissions and receptions, and the percentage of cooperating nodes. The energy savings in transmissions lies between 20% and 30%. Moreover, as the networks' densities increase, the amount of cooperating nodes slightly reduces, together with the energy saving achieved in transmissions and receptions. This suggests that VCB is even more appropriate for scenarios which involve sparse networks.

Other factor that can influence the cooperation is the path loss exponent. The greater this exponent, the higher the costs for transmissions. Fig. 6 shows the behavior of VCB when the path loss exponent varies. As expected, as the exponent grows, the higher is the energy economy of VCB, reaching 30% when $\alpha = 5$. This result suggests that VCB is highly recommended for noisy scenarios.

VCB is also able to establish cooperation when more than two networks are deployed in the same location, since its control is done locally. Fig. 7 shows the behavior of VCB when the number of distinct deployed networks increases (all with same configuration). The increase in the number of networks

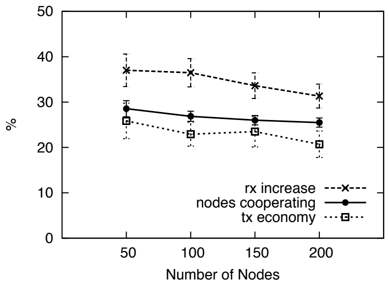


Fig. 5. Cooperation results when the number of nodes is varied.

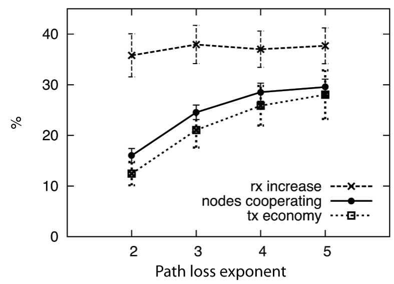


Fig. 6. Cooperation results when path loss exponent is varied.

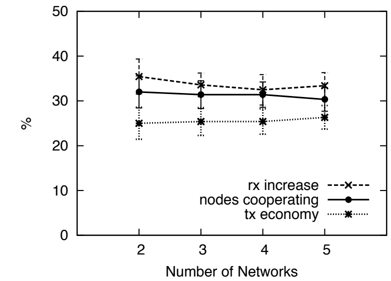
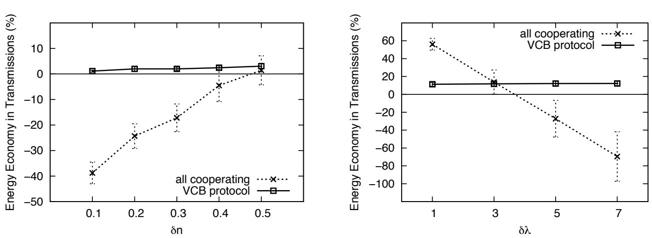


Fig. 7. Cooperation results when the number of networks is varied.

does not influence the three analyzed metrics. The network has approximately 30% of nodes cooperating, and with this cooperation is possible to save more than 20% of energy in transmission, with approximately 35% increase in receptions. These results show that VCB is scalable with the number of networks deployed.

The next scenario considers the case where the configurations of both networks are different. In the next results, the configuration of one network, N_1, is kept constant whereas the configuration of the other network, N_2, is varied. We analyze the performance of VCB in N_1 and compare it with a naive solution, where all nodes cooperate indiscriminately.

First, we vary the density of N_2 from 10% to 50% of nodes of N_1. We call the rate between the networks' densities δd. In Fig. 8-a, we verify that VCB is able to save energy of N_1 even when $\delta d = 10\%$. On the other hand, the naive solution harms N_1, making it consumes up to 40% more energy when cooperating. Second, we vary the number of events of N_2 according to parameter $\lambda_{Poisson}$, from 1 to 7. The $\delta\lambda$ is the difference among the number of events that are detected by the N_1 and N_2. Thus, when the $\delta\lambda$ is 3, the network N_1 detected three times more events than the network N_2. We see in the Fig. 8-b that VCB keeps a constant behavior, saving the energy of N_1 even when the event rate of the network N_2 increased. In contrast, when the all the nodes cooperate and the $\delta\lambda$ is 7, the energy consumption of N_1 is 60% higher.



(a) Networks have different densities. (b) Networks have different data collection rates.

Fig. 8. Cooperation results when the networks have different configurations.

VI. Conclusions

We have analyzed the problem of cooperation when two or more wireless sensor networks from different owners are deployed in the same area. The viability of cooperation is not trivial and should be achieved and maintained in an autonomous way by the networks, given the scalability problem of WSNs. Thus, we propose the Virtual Cooperation Bond (VCB) protocol as a distributed protocol that autonomously enables cooperation among different WSNs. First, we propose the *Give to Conquer* (GTC) strategy as a strategy of cooperation between two WSNs in a way that cooperation always brings benefits to both of the networks. Then, we propose the VCB as a model that implements the GTC strategy locally, in a way that cooperation may be autonomously achieved and maintained, independently of the networks' global configurations. Simulation results show that the VCB protocol reduces the energy consumption of the networks, therefore extending their lifetime, even in extremely heterogeneous scenarios, those where cooperation is not possible or beneficial when previous approaches present in the literature are considered.

Acknowledgments

This work is partially supported by CNPq and FAPEMIG.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey." *Computer Networks*, 38(4):393–422, 2002.
[2] G. Pottie and W. Kaiser, "Embedding the Internet Wireless Integrated Network Sensors," in *Communications of the ACM*, 43(5):51–58, 2000.
[3] D. Fudenberg and J. Tirole, *Game Theory*. MIT Press, 1991.
[4] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. Cambridge University Press, 2007.
[5] M. Felegyhazi, L. Buttyan, and J. P. Hubaux, "Cooperative Packet Forwarding in Multi-Domain Sensor Networks," in *Proc. of IEEE PerSeNS*, pp. 345–349, 2005.
[6] P. O. Vaz de Melo, F. D. da Cunha, J. M. Almeida, A. A. Loureiro, and R. A. Mini, "The Problem of Cooperation Among Different Wireless Sensor Networks," in *Proc. of ACM MSWiM*, pp. 86–91, 2008.
[7] K. Bicakci and B. Tavli, "Prolonging Network Lifetime with Multi-Domain Cooperation Strategies in wireless sensor networks," *Ad Hoc Netw.*, 8(6):582–596.
[8] M.-Y. Wu and W. Shu, "Intersensornet: strategic routing and aggregation," in *Proc. of IEEE GLOBECOM*, 5 pp., 2005.
[9] G. V. Crosby and N. Pissinou, "Evolution of Cooperation In Multi-Class Wireless Sensor Networks," *Proc. of IEEE LCN*, pp. 489–495, 2007.
[10] A. Tanenbaum and J. Wetherall, *Computer Networks*, 5th ed. Prentice Hall, 2010.
[11] D. Helbing and W. Yu, "The Outbreak of Cooperation Among Success-Driven Individuals Under Noisy Conditions," *Proc. of the National Academy of Sciences of the USA*, 106(10):3680–3685, 2009.
[12] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," in *Proc. of ACM MobiCom*, pp. 255–265, 2000.