

A Hybrid and Flexible Discovery Algorithm for Wireless Sensor Networks with Mobile Elements

Koteswararao Kondepu¹, Francesco Restuccia^{2,3}, Giuseppe Anastasi², Marco Conti³

¹Dept. of Computer Science & Engineering
IMT Institute, Lucca, Italy
koteswararao.kondepu@imtlucca.it

²Dept. of Information Engineering
University of Pisa, Italy
{g.anastasi, f.restuccia}@iet.unipi.it

³IIT-CNR
National Research Council, Italy
marco.conti@iit.cnr.it

Abstract—In sparse wireless sensor networks, data collection is carried out through specialized mobile nodes that visit sensor nodes, gather data, and transport them to the sink node. Since visit times are typically unpredictable, one of the main challenges to be faced in this kind of networks is the energy-efficient discovery of mobile collector nodes by sensor nodes. In this paper, we propose an adaptive discovery algorithm that combines a learning-based approach with a hierarchical scheme. Thanks to its hybrid nature, the proposed algorithm is very flexible, as it can adapt to very different mobility patterns of the mobile collector node(s), ranging from deterministic to completely random mobility. We have investigated the performance of the proposed approach, through simulation, and we have compared it with existing adaptive algorithms that only leverage either a learning-based or a hierarchical approach. Our results show that the proposed hybrid algorithm outperforms the considered adaptive approaches in all the analyzed scenarios.

Keywords: *Wireless Sensor Networks, Sparse Sensor Networks, Mobile Node Discovery, Energy Efficiency.*

I. INTRODUCTION

A *Wireless Sensor Network* (WSN) typically consists of a large number of sensor nodes, densely deployed over a geographical area. Sensor nodes are tiny devices that can acquire data from the surrounding environment, process them locally, and/or transfer them to a collection point (*sink node*) using multi-hop communication [1]. However, many real-life applications do not require a fine-grain sensing that necessitates such a dense deployment. Hence, a sparse topology can be used where sensor nodes are located at some strategic locations and the distance between neighboring nodes is typically much larger than their transmission range. In a sparse sensor network multi-hop communication is unfeasible, and data collection is carried out through *Mobile Elements* (MEs), i.e., special mobile nodes that visit sensor nodes regularly, gather data, and transport them to the sink node¹ [2, 3]. MEs can also be used in dense sensor networks to allow a more uniform distribution of energy consumption among sensor nodes, thus increasing the network lifetime [4]. Depending on the application scenario, MEs can be either part of the external environment (e.g., persons, cars, buses), or part of the system infrastructure (e.g., mobile robots). Also, they can

have very different mobility patterns, ranging from deterministic [5] to random mobility [6].

In WSNs with MEs (hereafter, WSN-MEs for shortness), the communication between sensor nodes and MEs is *opportunistic*, i.e., they can exchange data only when they are in the communication range of each other. In principle, a sensor node could be always in sleep mode and wake up only for communication. In practice, unless the ME's motion is deterministic, the sensor node cannot know in advance when the ME will enter its communication range. Hence, a *discovery protocol* is used for detecting the presence of the ME [3]. Discovery algorithms commonly used in WSN-MEs are based on *periodic listening*. In detail, the ME emits periodic beacons to announce its presence in the area, while sensor nodes wake up periodically (and for a short time) to listen for possible beacons. The period between two consecutive activations (*wakeup period*) of the sensor node should be as long as possible to minimize the energy consumed during the discovery phase. On the other side, using a too long activation period could compromise the effectiveness of the discovery process, i.e., contacts could be missed or detected very late (thus, leaving a short time available for data communication).

Typically, discovery protocols use *fixed* parameters (i.e., constant wakeup period and/or beacon emission rate [2, 4,6]). Better performance, in terms of energy efficiency, can be achieved through adaptive schemes based on learning techniques (for predicting the arrival time of the ME) [7,8], or relying on a hierarchical approach [9,10,11]. Learning-based algorithms are very well-suited when the ME's motion has some regularity that can be learned and exploited to predict the next arrival time with a certain accuracy. However, they are unsuitable when the ME moves without any regular pattern. Hierarchical discovery protocols typically require two different radios (namely, a wake-up radio and a data radio), which are not available in most of the existing sensor platforms. In addition, they are not able to learn and exploit information about the specific mobility pattern of the ME.

In this paper, we propose a hybrid discovery protocol (hereafter referred to as *Hybrid*) that combines both a learning-based approach and a hierarchical scheme. The proposed protocol is thus very flexible and can adapt to very different mobility scenarios. Unlike other hierarchical approaches, it does not require two different radios and, hence, it can be implemented in any sensor platform. We have evaluated the proposed protocol, by simulation, and compared it with other adaptive discovery schemes. The

¹ As a special case, the sink node itself can be mobile and play the role of mobile data collector.

obtained results show that our hybrid approach outperforms existing adaptive schemes that only leverage either a learning-based approach or a hierarchical scheme.

The rest of the paper is organized as follows. Section II presents the Hybrid protocol. Section III describes the simulation environment used for our analysis. Section IV compares the proposed protocol with other adaptive protocols. Finally, Section V concludes the paper.

II. HYBRID DISCOVERY PROTOCOL

As mentioned before, the proposed discovery protocol combines a learning-based approach with a hierarchical scheme. Specifically, it tries to learn the mobility pattern of the ME and predict the next arrival time, on the basis of the past history, using *Q-Learning* [12], i.e., a form of reinforcement learning that does not require a model of the environment. The duty cycle of the sensor node is then adjusted according to this prediction. Hence, the sensor node is in sleep mode for most of the time, and activates only when the ME is about to arrive. Since the prediction may not be accurate, the Hybrid algorithm exploits an additional hierarchical scheme to increase its energy efficiency. The sensor node initially activates with a *low duty cycle* and switches to a *higher duty cycle* only when the ME is actually nearby. Information about the physical location of the ME is made available to the sensor node by the ME itself by using two different Beacon messages, namely *Short Range Beacons (SRBs)* and *Long Range Beacons (LRBs)*. LRBs and SRBs are transmitted in an interleaved way, both with the same period (i.e., $2 \cdot T_{BI}$, so that the overall beaconing period is T_{BI}), but with different transmission power, and convey different information. SRBs are transmitted with the same transmission-power level used during the communication phase for data exchange. They experience a transmission range r – hereafter referred to as *communication range* – and are aimed at notifying the sensor node that the ME is within its transmission range and data exchange can, thus, take place. Instead, LRBs are sent with a higher transmission power. Therefore, they have a transmission range R larger than the communication range r – throughout R will be referred to as the *discovery range* – and are used to inform the sensor node that the ME is approaching and a contact could potentially occur shortly.

As mentioned before, the prediction algorithm is based on Q-Learning. Specifically, like RADA [8], it follows the *Distributed Independent Reinforcement Learning (DIRL)* approach [13], and relies on the following elements: (i) a state representation consisting of both system and application variables, (ii) a set of tasks (i.e. duty cycles) that can be executed by the sensor node, (iii) a reward function ρ associated with each task, and (iv) a utility function Q . The objective of the system is to maximize the long-term utility that can be achieved by executing the different tasks. In our system, the state s corresponds to the inter-contact time, as observed by the sensor node, i.e., the time elapsed from the beginning of a certain contact to the beginning of

the subsequent one. The reward function ρ provides the *immediate reward* achieved by executing a task. It is positive if a success has been obtained, and negative otherwise. Instead, the utility function gives the *long-term utility* of performing a task. Q is an utility look-up table whose generic element $Q(s, \tau)$ provides the utility of performing task τ in state s . It is defined as the expected value of the sum of the immediate reward ρ and the discounted utility of state s' resulting from execution of task τ , i.e.,

$$Q(s, \tau) = E[\rho + \gamma \cdot e(s') | s, \tau] \quad (1)$$

where $e(s') = \max_{\tau} Q(s', \tau)$. The expected value in equation (1) is conditioned to state s and task τ . Since Q-learning is done online, equation (1) cannot be applied directly, as the stored utility values may not have converged yet to their final values. In practice, Q-learning is used with incremental updates as given by the following equation:

$$Q(s, \tau) = (1 - \alpha) \cdot Q(s, \tau) + \alpha \cdot [\rho + \gamma \cdot e(s')] \quad (2)$$

In equation (2), α is a learning-rate parameter between 0 and 1, that controls the rate at which a sensor node tries to learn by giving more (α close to 1) or less (α close to 0) weight to the previously learned utility value. Furthermore, γ is a discount-factor, also between 0 to 1; the higher the value, the greater the sensor node relies on future reward rather than on immediate reward. Time is divided into time domains (of fixed duration T_D), and the utility function is updated periodically, at the end of each time domain. Then, based on the learned utility, the task that maximizes the long-term utility is selected for execution in the next future.

As any other learning algorithm, Hybrid includes both an *exploitation* and an *exploration* phase. During the exploitation phase, the next task is selected according to the learned utility (as described above), while in the exploration phase it is picked up *randomly* from the set of available tasks. The exploration phase is accessed, at the end of a time domain, with a probability ε evolving dynamically as

$$\varepsilon = \varepsilon_{\min} + \max\left\{0, \frac{(\varepsilon_{\max} - \varepsilon_{\min}) \cdot (c_{\max} - c)}{c_{\max}}\right\} \quad (3)$$

where ε_{\min} (ε_{\max}) is the minimum (maximum) exploration probability, while c and c_{\max} denote the number of contacts detected by the sensor node when equation (3) is evaluated, and the maximum number of detected contacts to be considered for calculating ε . Finally, there is a third phase (namely *activation* phase), triggered by the reception of a LRB from the ME, during which the next task to execute is chosen *deterministically* (see below). Although the definition of tasks is strictly related to the specific application, in Hybrid we defined the following tasks (i.e., duty cycles).

- *Sleep (SLP)*. The sensor node keeps the radio in sleep mode. This task is selected whenever the ME is not expected to arrive, based on the learned utility.
- *Low Duty Cycle (LDC)*. The sensor node operates with a low duty cycle δ_L . This task is selected when the ME is

expected to arrive in the next time domain, based on the learned utility.

- **High Duty-Cycle (HDC).** The sensor node operates with a high duty cycle δ_H . Unlike the other tasks, HDC is not selected on the basis of the learned utilities. It is chosen whenever a LRB is received from the ME (thus, starting the activation phase).

Algorithm 1: Hybrid algorithm

```

init
   $s = 0$ ;  $Q(0, \tau) = 0$  for all  $\tau$ ;
   $\Lambda = \{ \text{SLP, LDC, HDC} \}$ ;
  LRB-rcvd = False; SRB-rcvd = False;
   $T_{out} = (R + r)/v$ ;
  Select an initial task  $\tau$  from  $\Lambda$  randomly;
end init
loop
  execute  $\tau$ ;
  wait (event);
  switch (event) {
  case (LRB reception):
    LRB-rcvd = True;
     $\tau = \text{HDC}$ ; start timer( $T_{out}$ );
  case (timeout):
    LRB-rcvd = False;
     $\tau = \text{LDC}$ ;
  case (SRB reception):
    SRB-rcvd = True; stop timer;
    Start communication phase;
  case (end of communication):
    SRB-rcvd = False;
    LRB-rcvd = False;
     $\tau = \text{LDC}$ ;
  case (end of time domain):
    if SRB-rcvd = False {
      if (LRB-rcvd = True)  $\tau = \text{HDC}$ ;
      else {
        Calculate new state  $s'$ ;
        if  $\exists s'' : s' \approx s''$  then  $s' = s''$ 
        else add  $s'$  to the list of known states;
        Calculate reward for task  $\tau$  in state  $s$ ;
        Update  $Q(s, \tau)$ ;
        choose a new task  $\tau$  to execute
        // through exploration (with prob.  $\epsilon$ ) or exploitation
      }
    } // end if
  } // end switch
end loop

```

Algorithm 1 shows the actions performed by the sensor node. Initially, the algorithm initializes the look-up table Q and the set Λ of tasks that can be selected during the exploration phase (i.e., SLP, LDC, HDC). Boolean variables *LRB-rcvd* and *SRB-rcvd* are initialized to *False*. *LRB-rcvd* (*SRB-rcvd*) will be set when a LRB (SRB) is received, thus starting the activation (communication) phase. A node that has received a LRB may experience either a *contact* (if it

then receives a SRB) or a *false activation* (if it fails to receive a subsequent SRB). To avoid energy wastes due to false activations a timer is used. The timeout value T_{out} is set according to the worst case, i.e., when the distance between the sensor node and the ME is zero. Finally, an initial task is randomly selected from set Λ .

TABLE I. REWARD FUNCTION'S PARAMETERS.

LRB	SRB	n_c	p_m	e_p
NO	NO	0	-1	100
NO	YES	1	1	100
YES	YES	1	2	100
YES	NO	0	-2	100

At each step, the algorithm executes the previously selected task, until one of the following events occurs: (i) LRB reception; (ii) SRB reception; (iii) timeout expiration; (iv) end of the communication phase, and (v) end of current time domain. Upon receiving a LRB (case i) the sensor node sets the *LRB-rcvd* flag and selects the HDC task; finally, the false activation timer is started. If the latter timer expires without receiving any SRB (case ii), the sensor node selects the LDC as the next task and resets the *LRB-rcvd* variable. Instead, if a SRB is received before the timeout expiration (case iii), the sensor node sets the *SRB-rcvd* flag, stops the false activation timer, and enters the communication phase. At the end of the communication phase (case iv), both the *LRB-rcvd* and *SRB-rcvd* variables are reset. Finally, at the end of a time domain (case v), if the communication phase is in progress (i.e., a SRB has been received), no action is performed. If a LRB has been received (i.e., the sensor node is inside the activation phase), HDC is maintained as the next task. Otherwise, the new resulting state s' (i.e., inter-contact time) is measured. If s' is similar to a state s'' previously stored in the Q structure (i.e., the Hamming distance between s' and s'' is less than a pre-defined threshold [13]), s' is assimilated to s'' , otherwise s' is added to the list of known states. Finally, the reward for task τ corresponding to state s is calculated, and $Q(s, \tau)$ is updated accordingly. Specifically, the reward for any task is calculated as $\rho = (n_c \cdot p_m \cdot e_p - 1) \cdot e_s$, where n_c , p_m and e_p denote the number of contacts detected in the last time domain (i.e., 0 or 1), the price multiplier for the executed task, and the expected price, respectively. The negative part of the reward represents the cost for executing the task. This cost depends on to the time during which the sensor node was active during the last time domain (for instance, for the LDC task $e_s = \delta_L \cdot T_D \cdot P_{RX} + (1 - \delta_L) \cdot T_D \cdot P_{SL}$, where P_{RX} and P_{SL} denote the power consumption in receive mode and sleep mode, respectively). The reason behind using a price multiplier and an expected price is to allow a symmetric evaluation of the reward function. Thus, for each task, the reward is positive if the ME is successfully detected. If the ME is not detected, the reward is negative (equal to minus e_s). The price multiplier p_m for task τ is

calculated as shown in Table I.

III. SIMULATION ENVIRONMENT

To evaluate the performance of the proposed Hybrid protocol, we used the OMNeT++ simulation tool [14]. Without losing in generality, we considered a single ME moving with a constant speed v along a straight line at a fixed distance D from the sensor node. We focused on a *sparse* scenario, i.e., we assumed that the distance between neighboring sensor nodes is very large (i.e., larger than the discovery range R), so that we can concentrate on a single sensor node (the evaluation in a *dense* scenario can be found in [15]). In our analysis we measured the following performance indices.

- *Discovery Ratio*, defined as the ratio between the number of contacts successfully detected by the sensor node and the total number of potential contacts.
- *Activity Ratio*, defined as the ratio between the total active time (i.e. when the radio is on) and the overall time spent during the discovery phase.
- *Energy per Contact*, defined as the average energy consumed by sensor nodes in the discovery phase per detected contact.

The Discovery Ratio provides a measure of the *effectiveness* of a discovery scheme. Ideally, this index should be (close to) 100%. The activity ratio and the energy per contact measure the *energy efficiency* of the discovery scheme. Ideally, these indices should be as low as possible. To better evaluate the performance of Hybrid, we compared it with the following adaptive solutions that exploit either a learning-based or a hierarchical approach.

- *RADA* [8]. This protocol relies on the same prediction algorithm used in Hybrid, however, it does not exploit the hierarchical mechanism based on LRBs and SRBs. Sensor nodes are typically in sleep mode and activate only when the ME is expected to arrive.
- *2BD* [11]. This protocol uses a hierarchical approach based on LRBs and SRBs, but it is not able to predict the ME's arrival time. Sensor nodes are always in LDC and switch to HDC upon receiving a LRB.

For completeness, we also considered a fixed scheme (hereafter referred to as *Fixed*) where the duty cycle is constant over time and equal to HDC. Table 2 shows the duty cycle values used by the different protocols. To make the comparison fair, we considered the same values of HDC and LDC for the various algorithms. Also, we used the same set of duty cycles for Hybrid and RADA (i.e., HDC, LDC, SLP).

TABLE 2. DUTY CYCLE VALUES

Algorithm	HDC	LDC	SLP
HYBRID	3%	0.5%	0%
RADA	3%	0.5%	0%
2BD	3%	0.5%	-
Fixed	3%	-	-

In all the experiments, we performed 15 independent replications, each consisting of 1000 visits of the ME to the sensor node, and derived confidence intervals with a 90% confidence level. Since we are mainly interested in the discovery process, the channel quality was modeled using the disk model, i.e., packet loss is assumed to be 0% when the distance between sensor node and ME is lower than the communication range r , and 100% otherwise. Unless stated otherwise, all the other simulation parameters are as shown in Table 1. The learning parameters are set as in [8], while the power consumption values are derived from the datasheet of the Chipcon CC2420 radio transceiver [16].

TABLE 3. SIMULATION PARAMETERS.

Parameter	Value
LRB/SRB period ($2T_{Bl}$, Hybrid and 2BD)	200 ms
Beacon period (T_{Bl} , RADA and Fixed)	100 ms
Beacon duration (T_{BD} , all)	1 ms
ME Speed (v)	40 Km/h
Distance from the sensor node (D)	15 m
Communication range (r)	50 m
Nominal contact time	8.6 s
Discovery range (R , Hybrid and 2BD)	200m
Power Consumption in Receive Mode (P_{RX})	56.4 mW
Power Consumption in Sleep Mode (P_{SL})	60 μ W
Time Domain (T_D)	100 s
α (Hybrid and RADA)	0.5
γ (Hybrid and RADA)	0.5
ϵ_{max} , ϵ_{min} (Hybrid and RADA)	0.5, 0.05
c_{max} (Hybrid and RADA)	100

IV. SIMULATION RESULTS

A. Impact of the ME mobility pattern

In our analysis we assumed that the ME visits the sensor node at regular times, on average every T_{TOUR} (*inter-arrival time*). We considered four different ME mobility patterns resulting in a corresponding number of scenarios with increasing randomness in the inter-arrival time.

- *Deterministic*: ME arrivals are periodic. The inter-arrival time is fixed and equal to 30 min (1800s).
- *Gaussian-1*: The inter-arrival time is a random variable, distributed according to a normal distribution, with mean equal to 30 min and standard deviation equal to 1 min.
- *Gaussian-10*: As above, but with standard deviation equal 10 min.
- *Uniform*: The inter-arrival time is uniformly distributed in [0, 30] min.

Figures 1-3 show the impact of the ME's mobility pattern on the different discovery schemes, in terms of discovery ratio, activity ratio, and energy consumed per detected contact, respectively. When the mobility pattern is deterministic, all schemes exhibit a discovery ratio close to 100%. However, the activity ratio of Hybrid is significantly lower than that of the other schemes, resulting in a lower energy spent per detected contact.

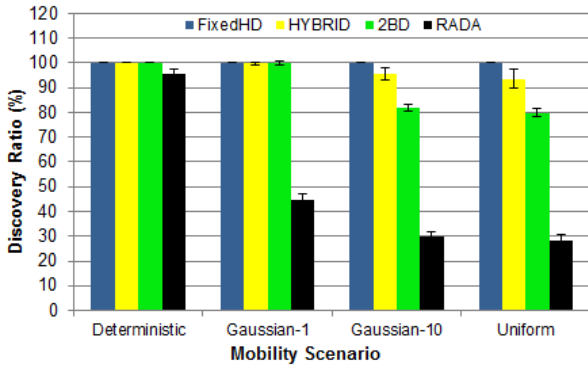


Figure 1. Impact of the mobility pattern in terms of discovery ratio.

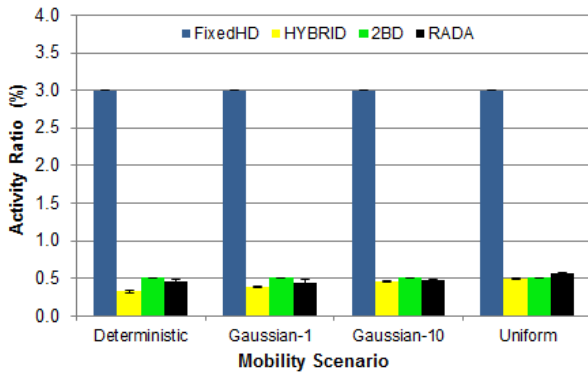


Figure 2. Impact of the mobility pattern in terms of activity ratio.

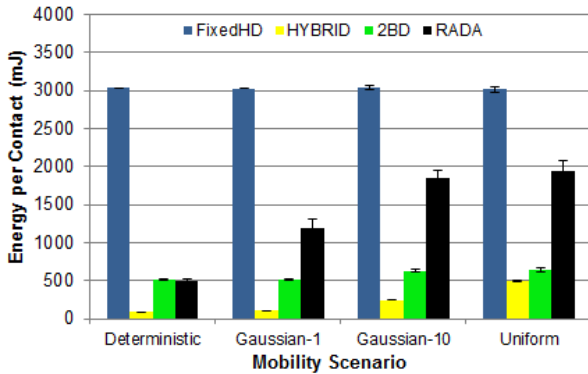


Figure 3. Impact of the mobility pattern in terms of energy per contact.

When the randomness of the inter-arrival time increases, the activity ratio of all the adaptive schemes is approximately the same (i.e., around 0.5%), meaning that the sensor node is in LDC for most of the time. However, Hybrid outperforms both 2BD and RADA in terms of percentage of detected contacts. Hence, it experiences a lower energy per contact. Specifically, Hybrid provides a discovery ratio very close to that of Fixed in all the considered mobility scenarios with an activity ratio of about 1/6 (in the worst case), thus achieving a huge reduction in energy consumption. For the sake of space, hereafter we will consider only the *Gaussian-1* scenario.

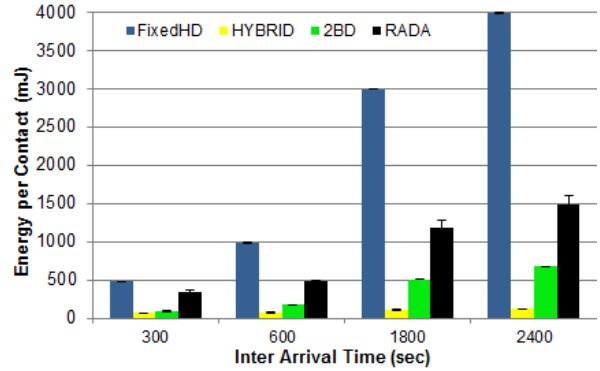


Figure 4. Impact of inter-arrival time terms of energy per contact.

TABLE 4. ENERGY SAVINGS PROVIDED BY HYBRID.

Inter-arrival Time (s)	2BD	RADA
300	27.9%	81.5%
600	59.2%	85.2%
1800	79.1%	91.0%
2400	82.6%	92.1%

B. Impact of the inter-arrival time

In Figure 3 the energy per contact has been calculated assuming an inter-arrival time of 1800s (i.e., 30 minutes). Obviously, the energy consumption is strongly influenced by this value. To investigate the impact of the inter-arrival time on the average energy consumed per detected contact, we considered different values for this parameter (while leaving all the other parameters unchanged). The obtained results, in terms of energy per detected contact, are summarized in Figure 4. As expected, the difference in the energy consumption of the Hybrid scheme, with respect to the other schemes, increases with the inter-arrival time. This is because the higher the inter-arrival time, the longer the time the sensor node spends in the discovery phase. Table 4, which shows the energy savings provided by Hybrid, with respect to 2BD and RADA, better emphasizes the benefits of using the proposed approach.

C. Impact of the Discovery Range

Both 2BD and Hybrid use a hierarchical mechanism based on LRBs and SRBs; LRBs are transmitted with a transmission range R , larger than the transmission range used for SRBs. It is thus extremely important to evaluate the impact of the R parameter on their performance. To this end, we considered three different values for R (i.e., 150m, 200m, 250m). The obtained results are shown in Figures 5-7. Fixed and RADA are not influenced by this parameter as they use a single beacon type. They have been included in the plots just for comparison. From our results it clearly emerges that increasing R increases the probability of detecting a potential contact. However, after a given values (200m in the considered scenario), a further increase in the R value does not produce any significant advantage in terms of discovery ratio, while it increases the energy

consumption. This behavior is better emphasized by the activity ratio, which tends to increase with R . This is because the sensor node remains in HDC for a time proportional to R . Finally, we can observe that Hybrid slightly outperforms 2BD for all the considered R values, in terms of activity ratio. This is because Hybrid can also exploit the prediction algorithm which puts the radio in sleep mode (selecting the SLP task) when there is a low probability to receive a LRB, based on the learning utility.

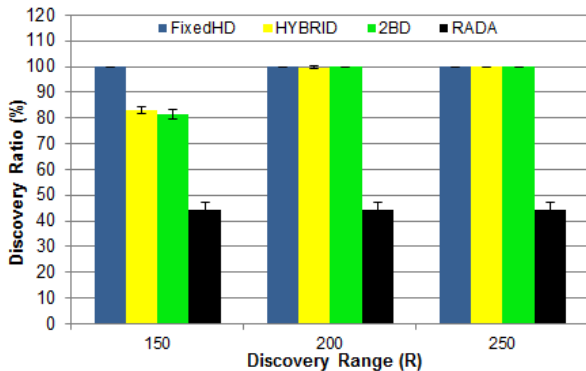


Figure 5. Impact of the Discovery Range on the discovery ratio.

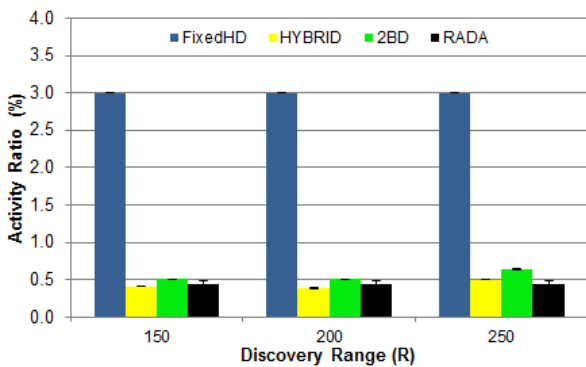


Figure 6. Impact of the Discovery Range on the activity ratio.

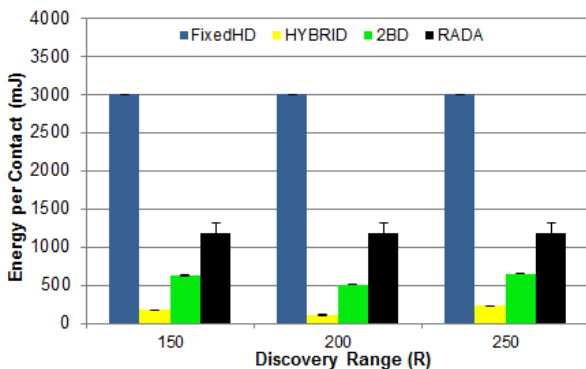


Figure 7. Impact of the Discovery Range on the energy per contact.

V. CONCLUSIONS

In this paper, we have proposed an adaptive discovery algorithm for WSNs with MEs, which combines a learning-

based approach with a hierarchical scheme. We have investigated the performance of the proposed algorithm, through simulation, in a sparse scenario. Our results show that, it can adapt to very different mobility patterns of the ME and, in comparison with other adaptive discovery algorithms, it allows a very large energy saving, especially when sensor nodes spend a long time in the discovery phase. For the sake of space, we only focused on a sparse scenario. The analysis in a dense scenario can be found in [16].

REFERENCES

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "Wireless Sensor Networks: a Survey", *Computer Networks*, Vol.38, N. 4, March 2002.
- [2] S. Jain, R. Shah, W. Brunette, G. Borriello, S. Roy, "Exploiting Mobility for Energy Efficient Data Collection in Wireless Sensor Networks", *Mobile Networks and Applications*, Vol. 11, No. 3, 2006.
- [3] M. Di Francesco, S. Das, G. Anastasi, "Data Collection in Wireless Sensor Networks with Mobile Elements: A Survey", *ACM Transactions on Sensor Networks*, Vol. 8, N.1, August 2011.
- [4] A. Somasundara, A. Kansal, D. Jea, D. Estrin, M. Srivastava, "Controllably Mobile Infrastructure for Low Energy Embedded Networks", *IEEE Trans. on Mobile Computing*, Vol. 5, N. 8, 2006.
- [5] A. Chakrabarti, A. Sabharwal, B. Aazhang, "Using Predictable Observer Mobility for Power Efficient Design of Sensor Networks. *Proc. International Workshop on Information Processing in Sensor Networks (IPSN 2003)*.
- [6] R. Mathew, M. Younis, S. Elsharkawy "Energy-Efficient Bootstrapping Protocol for Wireless Sensor Network", *Innovations in Systems and Software Engineering*, Vol. 1, No 2, Sept. 2005.
- [7] V. Dyo, C. Mascolo, "Efficient Node Discovery in Mobile Wireless Sensor Networks", *Lecture Notes in Computer Science. Proc. DCOSS 2008 (LNCS 5067)*. Springer, Heidelberg (2008).
- [8] K. Shah, M. Di Francesco, G. Anastasi, M. Kumar, "A Framework for Resource Aware Data Accumulation in Sparse Wireless Sensor Networks", *Computer Communications*, Vol. 34, N. 17, Nov. 2011.
- [9] J. Brown, J. Finney, C. Efstratiou, B. Green, N. Davies, M. Lowton, G. Kortuem, "Network Interrupts: Supporting Delay Sensitive Applications in Low Power Wireless Control Networks", *Proc. ACM Workshop on Challenged Networks (CHANTS 2007)*, Montreal, Canada, 2007.
- [10] H. Jun, M. Ammar, M. Corner, E. Zegura, "Hierarchical Power Management in Disruption Tolerant Networks Using Traffic-aware Optimization", *Computer Communications*, Vol. 32, N. 16, 2009.
- [11] K. Kondepudi, G. Anastasi, M. Conti, "Dual-Beacon Mobile-Node Discovery in Sparse Wireless Sensor Networks", *Proc. IEEE Symposium on Computers and Communications (ISCC 2011)*, Corfu, Greece, June 28 – July 1, 2011.
- [12] R. Sutton, "Temporal Credit Assignment in Reinforcement Learning", Dept. of Computer Science, Univ. of Massachusetts, Amherst, USA, COINS Technical Report 84-2 (1984).
- [13] K. Shah, M. Kumar, "Distributed Independent Reinforcement Learning (DIRL) Approach to Resource Management in Wireless Sensor Networks", *Proc. IEEE Conference on Mobile Ad hoc and Sensor Systems (MASS 2007)*, Pisa, Italy, 2007.
- [14] The OMNeT++ Network Simulator. <http://www.omnetpp.org>.
- [15] K. Kondepudi, F. Restuccia, G. Anastasi, M. Conti, "A Hybrid Discovery Algorithm for WSNs with Mobile Elements (Extended Version)", Technical Report DII-2012-4, Univ. of Pisa, April 2012. <http://www.iet.unipi.it/~anastasi/papers/tr-2012-4.pdf>
- [16] Chipcon, 2.4 GHz IEEE 802.15.4/ZigBee-Ready RF Transceiver, Chipcon Products from Texas Instruments, 2004, CC2420 Data Sheet.