Minimum-Energy Reprogramming with Guaranteed Quality-of-Sensing in Software-Defined Sensor Networks

Deze Zeng, Peng Li, Song Guo and Toshiaki Miyazaki
School of Computer Science and Engineering, The University of Aizu
Aizu-wakamatsu, Fukushima, Japan
{deze-z, pengli, sguo, miyazaki}@u-aizu.ac.jp

***Abstract*—After a decade of extensive research on application-specific wireless sensor networks (WSNs), the recent development of information and communication technologies make it practical to realize software-defined sensor networks (SDSNs), which are able to adapt to various application requirements and to fully explore the resources of WSNs. In SDSNs, wireless sensor nodes can be dynamically reprogrammed for different sensing tasks via the over-the-air-programming technique. For a given sensing task, it is usually required to guarantee certain quality-of-sensing, e.g., coverage ratio. Intuitively, the more sensors are deployed with a program, the higher quality-of-sensing of the corresponding task can be achieved. However, this is at the expense of high reprogramming energy consumption. In this paper, we investigate how to design an energy-efficient reprogramming strategy with guaranteed quality-of-sensing for a sensing task. To this end, two issues will be tackled: 1) the subset of sensors that shall be reprogrammed, i.e., reprogramming sensor selection and 2) the program distribution routing. They are jointly considered and formulated as an integer linear programming (ILP) problem, based on which an algorithm with low computation complexity is then proposed. The high efficiency of our algorithm is validated by extensive simulation studies.**

I. Introduction

Wireless Sensor Networks (WSNs) have been widely deployed for a wide span of applications including surveillance, tracking, and controlling. While many efforts have been made to enhance the applicability and performance of WSNs from different layers, they mainly consider the case that a WSN is dedicated for one sensing task. Such a single-task WSN is prone to (1) high deployment cost: multiple WSNs for respective tasks may be deployed in the same area, (2) low service reutilization: different vendors develop their WSNs in an isolated manner without sharing common functionalities, (3) difficult hardware recycling: altering existing code on single-task sensor nodes is difficult, highly error-prone, and costly.

Software-Defined Sensor Networks (SDSNs) emerge as a compelling solution to the above issues [1]. An SDSN consists of sensors whose functionalities can be dynamically configured by reprogramming [2]. Some SDSN prototypes have been practically realized. Miyazaki et al. [3], [4] implement software-defined sensor nodes that can dynamically change their sensing functions at runtime according to the sensing task requirements.

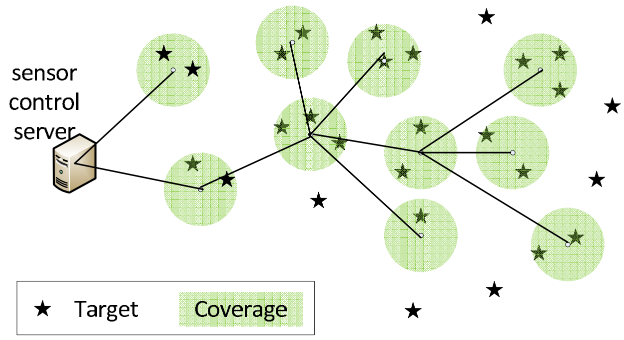

Fig. 1. An example of Software-Defined Sensor Network

In this paper, we consider an SDSN as shown in Fig. 1, which consists of one sensor control server and a set of software-defined sensor nodes. To deploy a new sensing task, the sensor control server shall reprogram some sensor nodes by distributing a corresponding program to them for the task. Only the reprogrammed sensors are able to sense the related targets within its coverage area. Since sensor nodes may be frequently reprogrammed, energy efficiency of reprogramming becomes an eye-catching issue in SDSN with battery-powered nodes. On the other hand, each task requires a certain level of quality-of-sensing, e.g., coverage ratio, which is a commonly adopted quality-of-sensing metric describing the portion of targets covered by the reprogrammed sensors [5]. Although reprogramming all sensors can always achieve the maximum coverage, it would lead to high energy consumption due to reprogramming unnecessary sensors for a given quality-of-sensing.

There are two factors that affect the reprogramming energy consumption: *reprogramming sensor selection* and *program distribution routing*. The former refers to selecting a subset of sensors to reprogram such that the required quality-of-sensing is achieved. The latter can be regarded as a multicasting that is to disseminate a program to the selected sensors. The minimum-energy multicasting problem has been widely discussed in the literature, e.g., [6], [7]. However, different from conventional multicasting where the destination set is known beforehand, to determine the reprogramming sensors for a sensing task with required quality-of-sensing is also a part of the problem. These two factors are highly correlated and shall be jointly considered. Therefore, in this paper, we are motivated to investigate how to minimize energy consumption on reprogramming by jointly considering sensor selection and multicasting routing under the constraints of quality-of-sensing. Our main contributions are summarized as follows:

- To our best knowledge, we are the first to study the minimum-energy reprogramming with guaranteed quality-of-sensing in SDSNs. We formulate it as an integer linear programming (ILP) problem by jointly considering sensor selection and multicasting routing.
- To deal with the high computational complexity of solving ILP, we propose an efficient heuristic algorithm based on the formulated ILP.
- Through extensive simulation based studies, we show the high efficiency of the proposed ILP-based algorithm. Furthermore, the advantage of joint optimization over the one considering two factors independently is also validated.

The rest of this paper is organized as follows. Section II provides a brief overview of related work. Section III gives our network model and problem statement. The ILP formulation and its derived heuristic are proposed in Sections IV and V, respectively. Section VI shows our performance evaluation results. Finally, Section VII concludes our work.

II. Related Work

A. Software-Defined Sensor Networks

It is widely agreed that SDSN is an inevitable developing trend of WSN with the fast development of various information and communication technologies [1]. A few pioneering investigations have been made in the literature. Rossi et al. [8] present their design and implementation of SYNAPSE++, a fountain codes based code dissemination strategy in WSNs. Miyazaki et al. [3], [4] recently realize Die-hard Sensor Network (DSN), which consists of sensor nodes that can dynamically adjust their sensing functions by turning on the corresponding sensing unit. Shafi et al. [9] propose an efficient over-the-air-programming technique called Queen's Differential that can detect and explore the similarities between the new problem and the older one to reduce the program size to be written. Our work is the first work that addresses the reprogramming energy consumption from the networking layer, to our best knowledge.

B. Minimum-Energy Multicasting

Wu et al. [6] show that under a layered model of wireless networks, the minimum energy-per-bit for multicasting in a mobile ad hoc network can be found by solving an ILP and the minimum energy-per-bit can be attained by performing network coding based multicasting. Lun et al. [7] consider the problem of establishing minimum-cost multicasting connections over coded packet networks, i.e., packet networks where the contents of outgoing packets are arbitrary, causal functions of the contents of received packets. In [10], the authors explore the benefits of power control for broadcast sessions and propose a heuristic algorithm called BIP (Broadcast Incremental Power) with a constant approximation ratio given in [11]. A formal NP-completeness proof for the energy-optimal broadcast problem is provided in [12]. The main difference between reprogramming and multicasting is on the awareness of the destination set, which is unknown in reprogramming. Furthermore, we will show that the destination set shall be jointly considered with the routing.

C. Coverage Problem in WSNs

To WSNs, an important issue is on the quality-of-sensing, e.g., coverage, which has been widely addressed in the literature from different aspects, e.g., placement, sensor scheduling, etc. Chakrabarty et al. [13] consider a problem on how to place sensor nodes in a grid to minimize the total cost and formulate it into an ILP problem, which is then solved by a divide-and-conquer method. Cheng et al. [14] address the minimum coverage beach problem NP-Complete under bandwidth constraints. It is then formulated into an ILP problem and solved using heuristic algorithms. Slijepcevic et al. [15] present a SET-K cover problem in which sensors are organized into mutually exclusive sets and each set is equivalent to a subarea monitored by a corresponding sensor. Huang et al. [16] discuss the way to determine how well a sensor network is monitored or tracked by sensors by formulating two decision problems for unit-disk coverage and non-unit disk coverage, respectively. Recently, Kasbekar et al. [17] design a polynomial-time distributed algorithm for maximizing the lifetime of the network by designing a schedule on the sequence of sensors that shall be activated in every time slot. The key difference of SDSNs to conventional single-task WSNs is on the reprogrammability of sensors. The coverage issue is also highly related to the reprogramming energy consumption. Our work takes the coverage requirement as an input to the minimum-energy reprogramming problem.

III. System Model and Problem Statement

In this paper, we consider an SDSN as shown in Fig. 1, which can be modeled as a directed graph $G = (N, E)$. The vertex set N includes one control server s and a set I of randomly distributed sensor nodes using fixed transmission power. Any direct transmission link (m, n) is included in the arc set E if receiver n is within the transmission range of sensor m.

We consider a set K of sensing targets that are randomly distributed within the same region of the SDSN. When a sensor node is reprogrammed to conduct a sensing task, it covers a set of targets within its sensing range. The coverage relationship between a sensor node $i \in I$ and target $k \in K$ can be expressed by:

$$v_{ik} = \begin{cases} 1, \text{ if target } k \text{ is within the sensing range of } i \ , \\ 0, \text{ otherwise.} \end{cases}$$

In this paper, we adopt coverage ratio as the quality-of-sensing metric. To guarantee the quality-of-sensing required by a sensing task, a portion δ of targets must be covered by a set of reprogrammed sensor nodes.

The random linear network coding is adopted to disseminate the program, which is divided into batches of coded packets, from the control server to a set of chosen sensors. At the control server, a coded packet a'_i is a linear combination of H native packets $a_j (1 \leq j \leq H)$: $a'_i = \sum_{j=1}^{H} \alpha_{ij} a_j$, where α_{ij} is the coding coefficient chosen from a Galois Field (for example, $GF(2^8)$). At any intermediate node, it is created by randomly mixing the coded packets in the buffer as $a''_i = \sum_j \beta_{ij} a'_j$, which is still a linear combination of H native packets if β_{ij} is chosen from the same Galois Field. The node receiving a packet successfully first checks whether it is linearly independent with the coded packets already in its buffer. If it is independent, this innovative packet is inserted into its buffer. Otherwise, it is discarded. A sensor node can decode the original programs once it receives H linearly independent coded packets.

For a given sensing task, our objective is to minimize the energy consumption by reprogramming a set of sensors to achieve the required quality-of-sensing. The reprogramming process involves determining a set of sensors to be reprogrammed and the multicasting routing for distributing the program to these sensor nodes. Since it is not necessary to reprogram all sensor nodes for a given coverage ratio, it is essential to find out an appropriate set of sensors to save energy consumption. On the other hand, the program distribution can be viewed as a multicasting process. The larger the destination set, potentially the higher the energy consumption since more forwarding operations may be required to reach all the destinations. Furthermore, sensing sensor selection and multicasting routing are highly correlated with each other and therefore shall not be considered independently.

IV. Problem Formulation

In this section, we formulate minimum-energy reprogramming problem with the joint consideration of sensor selection and multicasting routing.

A. Quality-of-sensing Constraints

We first consider the quality-of-sensing requirement for a sensing task in SDSNs. A sensor $i \in I$ can sense target $k \in K$ if and only if 1) target k is within the sensing range of i, i.e., $v_{ik} = 1$, and 2) it is reprogrammed for the sensing task. We define a binary variable α_i to denote whether sensor i is selected to be reprogrammed or not as follows:

$$\alpha_i = \begin{cases} 1, \text{ if sensor } i \text{ is reprogrammed,} \\ 0, \text{ otherwise.} \end{cases}$$

By letting β_{ik} denote whether sensor i is able to sense target k, we have

$$\beta_{ik} = \alpha_i v_{ik}, \forall i \in I, k \in K. \quad (1)$$

A sensing target is covered only when there is at least one sensor can sense for it. We use a binary value $\gamma_k \in \{0, 1\}$ to represent whether target k is covered or not. Then, we have:

$$\sum_{i \in I} \beta_{ik}/I \leq \gamma_k \leq \sum_{i \in I} \beta_{ik}, \forall k \in K. \quad (2)$$

The correctness of (2) can be verified as: 1) $\gamma_k \equiv 1$ if $\exists i \in I, \beta_{ik} = 1$ and 2) $\gamma_k \equiv 0$ if $\forall i \in I, \beta_{ik} = 0$

To ensure the quality-of-sensing for a sensing task, a coverage ratio δ shall be achieved. This leads to the following constraint:

$$\sum_{k \in K} \gamma_k / K \geq \delta. \quad (3)$$

B. Network Flow Constraints

Since multiple sensors should be reprogrammed to achieve a given coverage ratio, the program distribution process can be treated as a multicasting process. In this paper, we propose a multicasting scheme using network coding that has shown a lot of advantages in achieving high throughput and energy-efficiency. Let $f^i_{(m,n)}$ denote the flow disseminating to sensor node i over link $(m, n) \in E$. We have the following constraints for flow conservation:

$$\sum_{(m,n) \in E} f^i_{(m,n)} - \sum_{(n,v) \in E} f^i_{(n,v)} = \sigma^i_n, \forall i \in I, n \in N, \quad (4)$$

where

$$\sigma^i_n = \begin{cases} -\alpha_i H, \text{ if } n = s \\ \alpha_i H, \text{ if } n = i \\ 0, \text{ otherwise,} \end{cases}$$

and H is the size of the program to be distributed. Note that different from conventional multicasting where all the destinations are known beforehand, any sensor node could be a destination node in our case. According to (4), we can see that whether a sensor node shall be reprogrammed is determined by the value of α_i. If $\alpha_i = 1$, σ^i_i equals to H, indicating that a flow of size H shall be disseminated from source s to sensor i. Otherwise, node i serves as a forwarder in the network.

Due to network coding, the actual total flow f_m emitted from m should satisfy

$$f_m \geq \sum_{(m,n) \in E} f^i_{(m,n)}, \forall i \in I, m \in N. \quad (5)$$

Let η be the energy cost per data unit. The total energy cost C for the whole program distribution process can be calculated by summing up the energy consumption on all the nodes, i.e.

$$C = \sum_{m \in N} \eta f_m. \quad (6)$$

C. Problem Formulation

By summarizing all the above constraints together, we obtain an integer linear programming (ILP) as:

$$\begin{aligned} \min : & \sum_{m \in N} \eta f_m, \\ \text{s.t.} : & (1), (2), (3), (4), (5), \\ & \alpha_i \in \{0, 1\}, \beta_{ik} \in \{0, 1\}, \gamma_k \in \{0, 1\}, \forall i \in I, k \in K. \end{aligned}$$

Note that the reprogramming sensor selection with guaranteed quality-of-sensing and the multicasting routing towards minimum-energy program distribution are jointly considered in the formulation above.

V. An ILP-based Algorithm

Since it is computationally prohibitive to solve the ILP problem to get the optimal solution in large-scale sensor networks, we propose a computation-efficient heuristic algorithm in this section. Our main idea is to first find a set of sensors that should be reprogrammed by solving a relaxed ILP problem, and then determine the optimal multicasting flow by setting these found sensors as destinations.

The overall ILP-based algorithm is presented in Algorithm 1. We first relax all the integers and solve the resulting linear programming (LP) problem. Note that all the solutions are floating values, including the reprogramming sensor selection values $\alpha_i, \forall i \in I$. Next, we try to find a set of sensors that achieve the required coverage by rounding the floating variable α_i. Intuitively, the one with the highest value shall be converted with the highest priority. Therefore, we first sort all $\alpha_i, \forall i \in I$ in a decreasing order into set A in line 2. Then, we try to convert the sorted α_i into binary until the required coverage ratio is guaranteed, between lines 3 and 18. Starting from the first element in A, i.e., $j = 0$ (line 3), we iteratively set the jth elements into 1 and the rest as 0. With the temporary binary values of $\alpha_i, \forall i \in I$, the coverage relationship γ_k for all elements can be calculated by (1) and (2).

Note that not all conversions are beneficial to the coverage. To avoid unnecessary energy consumption to the sensor nodes that cannot contribute to the coverage, we further check whether the coverage ratio is improved after the conversion to exclude these nodes. As shown in lines from 11 to 16, we first calculate the new coverage ratio with the updated γ_k in line 11. Next, only when the coverage ratio is improved, i.e., $cr_{new} > cr_{old}$, we reserve current conversion and update the value of cr_{old}. Otherwise, we convert the jth element in A to 0.

The above conversion process proceeds until the coverage ratio exceeds the quality-of-sensing requirement. The binary values of $\alpha_i, \forall i \in I$ can be thus determined. With the determined destination set, we then integrate them into the network flow based multicasting model to obtain the optimal routing solution of $f_m, \forall m \in N$.

Algorithm 1 An ILP-based Algorithm

Input: reachability $v_{ik}, i \in I, k \in K$, graph $G = \{N, E\}$
Output: reprogramming sensor selection $\alpha_i, \forall i \in I$, routing flow $f_m, \forall m \in N$

1: Relax the integer variables in the ILP, and solve the resulting linear programming.
2: Sort $\alpha_i, i \in I$ decreasingly into set A
3: $j \leftarrow 0$
4: $cr_{old} \leftarrow 0$
5: **while** $cr_{old} < \delta$ **do**
6: $\quad A[j] \leftarrow 1$
7: $\quad A[u] \leftarrow 0, \forall u = [j, I]$
8: $\quad$ **for all** $k \in K$ **do**
9: $\quad\quad$ Calculate γ_k by (1) and (2)
10: $\quad$ **end for**
11: $\quad cr_{new} \leftarrow \sum_{k=1}^{K} \gamma_k / K$
12: $\quad$ **if** $cr_{new} > cr_{old}$ **then**
13: $\quad\quad cr_{old} \leftarrow cr_{new}$
14: $\quad$ **else**
15: $\quad\quad A[j] \leftarrow 0$
16: $\quad$ **end if**
17: $\quad j \leftarrow j + 1$
18: **end while**
19: Construct an LP problem by integrating fixed α_i as

$$\min_{f_m} : \sum_{m \in N} \eta f_m,$$
$$\text{s.t.} : (4), (5)$$

20: Solve above LP, and get the solution of f_m

VI. Performance Evaluation

In this section, we present our simulation-based performance evaluation results on the efficiency of our proposed algorithm. Algorithm 1 is implemented in our simulator. Besides, we also realize a two-phase algorithm that independently considers reprogramming sensor selection and program distribution routing as a comparator. The two-phase algorithm is designed with the concept that less energy shall be consumed if less number of sensors shall be reprogrammed. Therefore, in the first phase, the minimum number of sensors that shall be reprogrammed to guarantee the quality-of-sensing is found. Then, in the second phase, by taking the values of $\alpha_i, \forall i \in I$ determined in the first phase to obtain program distribution routing solution $f_m, \forall m \in N$. Furthermore, the optimal results "Optimal" are obtained by solving the ILP problem using commercial solver Gurobi optimizer [18].

We consider a 100×100 sensor network area. We fix the program size H as 10 and normalize the energy consumption per data unit η as 1. The values of coverage ratio requirement δ, sensing range r, communication range l and network size $|I|$ are varied in different group of simulations to extensively investigate the performance of both algorithms. 50 simulation instances in each group are conducted to get the average energy consumption. In each simulation instance, the positions of both sensor nodes and sensing targets are randomly generated. The sensor control server is always fixed at point (50, 50). Such position information is then applied into Algorithm 1, two-phase algorithm and (7) to get the energy consumption of "ILP-based", "Two-phase" and "Optimal", respectively.

A. Evaluations on the Reprogramming Energy Consumption

Let us first check how the coverage ratio requirement affects the reprogramming energy consumption. In this group of simulations, we fix $|I| = 50, |K| = 30, H = 10, r = 10, l = 30$ and vary δ from 0 to 1.0. Fig. 2 presents the simulation results. We notice that the performance of our ILP-based algorithm approaches the optimal one and much outperforms the one of two-phase algorithm. For example, when $\delta = 0.4$, the total reprogramming energy consumption is 21.3, 24.1 and 38.4 for "Optimal", "ILP-based" and "Two-phase", respectively. This not only verifies the high energy efficiency of Algorithm 1

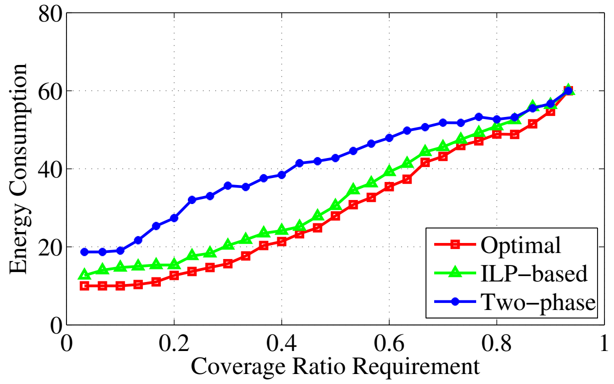

Fig. 2. On the effect of coverage ratio requirement

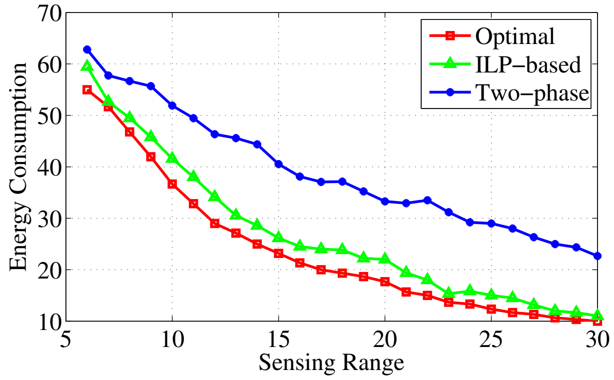

Fig. 3. On the effect of sensing range

but also shows the advantage of joint consideration of reprogramming sensor selection and program distribution routing. It implies that less sensors to be programmed does not mean lower reprogramming energy consumption. Reprogramming sensor selection shall be considered in conjunction with program distribution routing.

We further observe the energy consumption is a non-linear increasing function of δ. The energy consumption first increases slowly and then becomes fast with the increasing of δ. Such phenomenon is attributed to the following factors. When the required coverage ratio δ is small, few sensors near the sensor control server s are able to guarantee the quality-of-sensing. Later, when δ becomes high, not only the reprogramming sensor set is enlarged, but also the average programming forwarding hops are raised since many sensors far away from s need to be reprogrammed to satisfy the coverage ratio requirement. Many forwarding operations are required in this case. Finally, when δ is quite big, e.g., $\delta = 0.93$ in Fig. 2, almost all the sensors that can reach some sensing targets shall be reprogrammed and all the three algorithms converge to the same performance since no much optimization space left for both "Optimal" and "ILP-based".

The sensing ability, e.g., sensing range r, of the sensor nodes shall also influence the energy consumption. Fig. 3 shows the evaluation results on the effect of sensing range to the energy consumption by fixing δ as 0.6 and varying r from 7 to 30. Once again, we notice that "ILP-based" outperforms

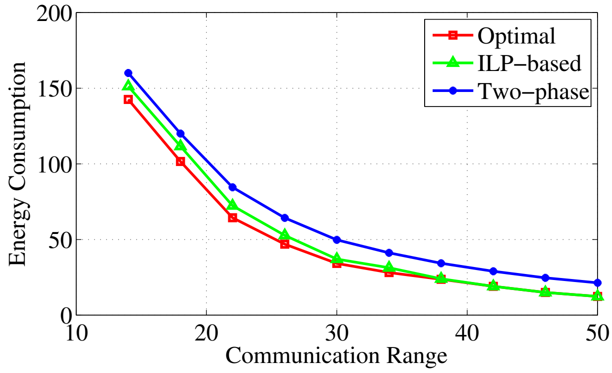

Fig. 4. On the effect of communication range

"Two-phase" algorithm and approaches to the optimal solution under any sensing range. Furthermore, the energy consumption shows as a non-linear decreasing function of the sensing range. When the sensing range is small, each sensor node can reach only a limited number of targets and therefore many sensors must be reprogrammed to satisfy the quality-of-sensing requirement. Consequently, many forwardings are required and high energy is consumed. The non-linearity comes from the fact that the coverage area, and hence the number of targets, increases quadratically with the sensing range.

Next, we study the effect of the communication range l to the energy consumption. In this group of evaluations, we fix the values of $\delta = 0.6, r = 10$ and vary l from 17 to 50. The results are shown in Fig. 4, from which we can see that the energy consumption also shows as a non-linearly decreasing function of the communication range. This is due to the fact that the longer the communication range is, the more sensors can be reached upon each forwarding in wireless communications with broadcast nature. Less transmission energy shall be consumed to deploy the programs onto the selected sensors. Similarly, the non-linearity is because the number of sensors nodes that can be communicated directly with a sensor increases quadratically with the the l.

Finally, Fig. 5 gives the results on the effect of network size $|I|$ under the values of $|K| = 50, \delta = 0.6, r = 10, |l| = 30$ and $|I|$ varied from 50 to 100. It can be noticed that the energy consumption shows as a decreasing function of network size $|I|$. This is because with more sensor nodes, there are more options to find appropriate nodes near the sensor control server to satisfy the quality-of-sensing requirement and thus less reprogramming energy will be consumed. Contrarily, when network size is small, some sensors selected to be reprogrammed may be far away from the sensor control server in this case and multiple forwardings are required to distribute the program to them. High reprogramming energy is thus consumed. Nevertheless, under any network size, we can always see the closeness of ILP-based algorithm to the optimal one and the huge advantage over "Two-phased" algorithm.

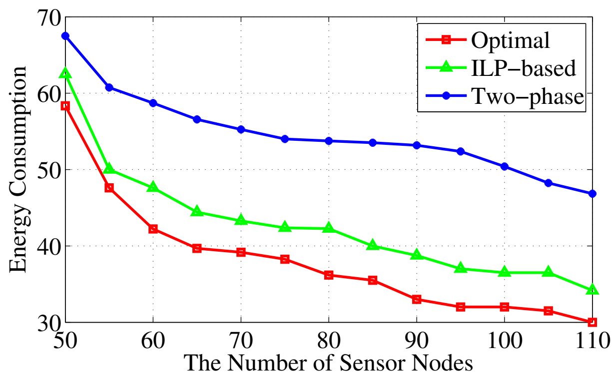

Fig. 5. On the effect of sensing network size

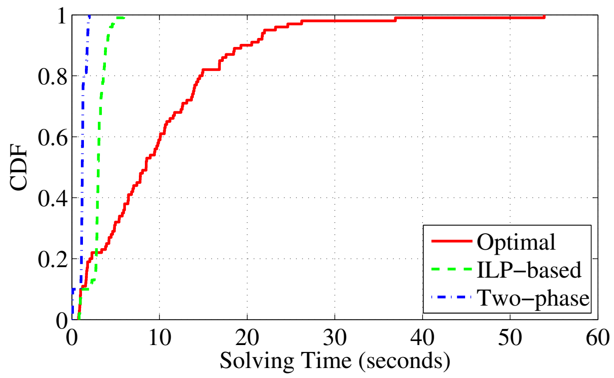

Fig. 6. On the computation time cost of algorithms

B. Evaluations on the Algorithm Solving Time

Besides the performance evaluations on reprogramming energy consumption, we are also interested in the computation time cost of each algorithm. To this end, we conduct 100 simulations under the settings $|I| = 50, |K| = 30, r = 10, \delta = 0.4, l = 30$ and collect the computation time cost for each instance, based on which the cumulative distribution function (CDF) of solving time (in seconds) is derived, as shown in Fig. 6. We first notice that our ILP-based algorithm has similar computation time as Two-phase algorithm. For most cases, roughly more than 80%, both ILP-based and Two-phase algorithms can solve the problem within 5 seconds. However, to obtain the optimal solution by solving the ILP formulation, the solving time may be as high as 53.90 seconds in our tested cases. Remember that our algorithm can approach the optimal one on the reprogramming energy consumption. The high efficiency of our ILP-based algorithm is thus validated.

VII. Conclusion

In this paper, we investigate the problem of minimum-energy reprogramming with guaranteed quality-of-sensing in SDSNs. The reprogramming sensor selection on which sensor nodes shall be reprogrammed and the program distribution routing of the corresponding program to the selected sensors are jointly studied. We novelly describe the problem using an ILP formulation, based on which a computation-efficient algorithm is also proposed. By extensive simulation-based evaluations, we discover that our ILP-based algorithm performs close to the optimal one. By comparing against a two-phase algorithm that considers the reprogramming sensor selection and program distribution routing independently, the high efficiency and advantage of the joint consideration of the two issues is validated by the fact that our ILP-based algorithm much outperforms two-phase algorithm.

Acknowledgment

This work is partly supported by Strategic Information and Communications R&D Promotion Programme (SCOPE No. 121802001) and Grants-in-Aid for Scientific Research (No. 23500095).

References

[1] T. Luo, H.-P. Tan, and T. Quek, "Sensor OpenFlow: Enabling Software-Defined Wireless Sensor Networks," *IEEE Communications Letters*, vol. 16, no. 11, pp. 1896–1899, 2012.

[2] D. Zeng, T. Miyazaki, S. Guo, T. Tsukahara, J. Kitamichi, and T. Hayashi, "Evolution of Software-Defined Sensor Networks," in *Proc. of the 9th MSN*. IEEE, Dec 2013, pp. 410–413.

[3] T. Miyazaki, D. Shitara, Y. Endo, Y. Tanno, H. Igari, and R. Kawano, "Die-hard Sensor Network: Robust Wireless Sensor Network Dedicated to Disaster Monitoring," in *Proc. of the 5th ICUIMC*. ACM, 2011, pp. 53:1–53:10.

[4] T. Miyazaki, "Dynamic Function Alternation to Realize Robust Wireless Sensor Network," *International Journal of Handheld Computing Research*, vol. 3, no. 3, pp. 17–34, Jul. 2012.

[5] B. Wang, "Coverage Problems in Sensor Networks: A Survey," *ACM Computer Survey*, vol. 43, no. 4, pp. 32:1–32:53, Oct. 2011.

[6] Y. Wu, P. A. Chou, and S.-Y. Kung, "Minimum-Energy Multicast in Mobile Ad hoc Networks using Network Coding," *IEEE Transactions on Communications*, vol. 53, no. 11, pp. 1906–1918, 2005.

[7] D. S. Lun, N. Ratnakar, M. Médard, R. Koetter, D. R. Karger, T. Ho, E. Ahmed, and F. Zhao, "Minimum-Cost Multicast Over Coded Packet Networks," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2608–2623, 2006.

[8] M. Rossi, N. Bui, G. Zanca, L. Stabellini, R. Crepaldi, and M. Zorzi, "SYNAPSE++: Code Dissemination in Wireless Sensor Networks Using Fountain Codes," *IEEE Transactions on Mobile Computing*, vol. 9, no. 12, pp. 1749–1765, 2010.

[9] N. Bin Shafi, K. Ali, and H. Hassanein, "No-reboot and Zero-Flash Over-the-air Programming for Wireless Sensor Networks," in *Proc. of the 9th SECON*, 2012, pp. 371–379.

[10] J. Wieselthier, G. Nguyen, and A. Ephremides, "On the Construction of Energy-efficient Broadcast and Multicast Trees in Wireless Networks," in *Proc. of the 19th INFOCOM*, 2000, pp. 585–594.

[11] P.-J. Wan, G. Calinescu, X.-Y. Li, and O. Frieder, "Minimum-Energy Broadcast Routing in Static Ad Hoc Wireless Networks," in *Proc. of the 20th INFOCOM*, 2001, pp. 1162 –1171.

[12] M. Čagalj, J. Hubaux, and C. Enz, "Minimum-energy Broadcast in All-wireless Networks: NP-Completeness and Distribution Issues," in *Proc. of ACM MobiCom*, 2002, pp. 172–182.

[13] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho, "Grid Coverage for Surveillance and Target Location in Distributed Sensor Networks," *IEEE Transactions on Computers*, vol. 51, no. 12, pp. 1448–1453, 2002.

[14] M. X. Cheng, L. Ruan, and W. Wu, "Achieving Minimum Coverage Breach under Bandwidth Constraints in Wireless Sensor Networks," in *Proc. of the 24th INFOCOM*, vol. 4. IEEE, 2005, pp. 2638–2645.

[15] S. Slijepcevic and M. Potkonjak, "Power Efficient Organization of Wireless Sensor Networks," in *Proc. of ICC*, vol. 2. IEEE, 2001, pp. 472–476.

[16] C.-F. Huang and Y.-C. Tseng, "The Coverage Problem in a Wireless Sensor Network," in *Proc. the 2nd WSNA*. ACM, 2003, pp. 115–121.

[17] G. S. Kasbekar, Y. Bejerano, and S. Sarkar, "Lifetime and Coverage Guarantees Through Distributed Coordinate-Free Sensor Activation," *IEEE/ACM Transactions on Networking*, vol. 19, no. 2, pp. 470–483, 2011.

[18] Gurobi Optimization, "Gurobi Optimizer Reference Manual," 2013. [Online]. Available: http://www.gurobi.com