SATA: An Intelligent Security Aware Task Allocation for Multihop Wireless Networks

Wanli Yu*, Ojo Akinlolu*, Yannik Sander*, Yanqiu Huang† and Alberto Garcia-Ortiz*
* University of Bremen, Germany, {wyu, ojoak, ysander, agarcia}@uni-bremen.de
†University of Twente, Netherlands, y.huang-3@utwente.nl

***Abstract*—Multihop wireless networks, which consist of sets of battery powered wireless nodes, have been widely spreading in numerous IoT applications. As the nodes typically have limited resources, many energy aware task allocation schemes are conducted to achieve energy efficiency. However, the security concerns should be also considered, due to the vulnerable characteristics of wireless communication. This work addresses the energy and security concerns simultaneously. It proposes an intelligent security aware task allocation algorithm (SATA) based on genetic algorithm, to optimize the energy consumption while fulfilling the security requirements of both the application and the surrounding environment of sensor nodes. The extensive simulation results demonstrate significant improvement in various testing environments.**

I. Introduction

Wireless multihop networks have been widely spreading in numerous IoT scenarios [1]. The battery powered sensor nodes not only function as independent processing units, but also can collaborate together to complete the applications. Due to the limited resources of the sensor nodes and the vulnerable characteristics of the wireless medium, energy efficiency and security guarantee are two fundamental concerns.

Numerous energy aware task allocation schemes have been conducted to achieve the energy efficiency by distributing the workload for each node in the network. The authors in [2] propose an optimal task allocation algorithm to maximize the lifetime of cluster based wireless network. They formulate the task allocation problem as a linear programming problem (LP). DOTAM is proposed by [3] to optimize the energy consumption and real-time performance of multihop mesh networks. Both LP based centralized and Dantzig-Wolf decomposition based distributed algorithms are presented to approach the optimal solutions. However, the above task allocation schemes have not considered the security requirements, e.g., data confidentiality in communication both between the tasks and between the sensor nodes. In [4], the authors present a Tabu search based heuristic algorithm to minimize the energy consumption for safety critical applications. The security levels are used by applying different cryptographic algorithms. Similarly, a genetic algorithm (GA) based heuristic method is proposed in [5]. It uses a level-based modeling of cryptographic algorithms using mixed cryptographic implementations. Nevertheless, these security aware methods are designed for traditional embedded systems and they have not considered the affections of the multihop wireless medium. Consequently, it is essential to design security aware task allocation considering the multihop identity for the spreading wireless multihop networks.

This work integrates the security guarantee with the task allocation to optimize the energy consumption while fulfilling the security requirements. As GA is popular in optimization problems for the easy implementation and high probability of finding an optimal solution [6], we propose an intelligent **s**ecurity **a**ware **t**ask **a**llocation algorithm (SATA) based on GA for multihop wireless networks. To the best of our knowledge, this is the first work considering the special multihop wireless medium for the optimization of security aware task allocation. The rest of this paper is organized as follows. Section II presents the system models. The next section illustrates the proposed SATA algorithm. Section IV presents the evaluation results. The last section summarizes this work.

II. System Models

This section presents the system models including the network structure, the application and security models, and the cost functions of the wireless nodes.

A. Network Structure and The Application Model

The network is made up of a set of wireless sensor nodes, $N_1, \cdots, N_n$, and one sink node. They are connected by multiple wireless hops and collaborate together to periodically execute the given application. In addition to sense the data and process the sensed data, the sensor nodes can also serve as the routing nodes to forward the data or share part of the tasks of the given application. The routing path is built based on the given routing algorithms. In this work, we employ a minimum hop routing algorithm [7].

The application of a wireless network is modeled as a Directed Acyclic Graph (DAG), $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$, as in [2], [8]. Each vertex $v \in \boldsymbol{V}$ represents a task of the application and each edge $\varepsilon \in \boldsymbol{E}$ stands for the communication between each pair of connected tasks. A task can be executed only when it receives the input data from all predecessor tasks.

B. Security Model

This work focuses on protecting the confidentiality of the data communication over the vulnerable wireless medium. We consider that the intra-node data communication is secure. Different security levels are used to indicate the strengths of the cryptography algorithms for the transmitted data. The security level is affected by the selected encryption algorithms, the number of encryption rounds and the key sizes, and the higher level corresponds to stronger security and heavier computation [4], [5]. Note that this work considers the security requirements of both the applications and the surrounding environment of sensor nodes. The communication between

978-1-7281-6687-2/20/$31.00 ©2020 IEEE

each pair of tasks has to meet the initial application security requirements. Due to the fact that the sensor nodes may be distributed in different physical environment, the security requirements for each pair of sensor nodes should be also considered. Considering the example in Fig. 1: v_1, v_2 and v_3 are 3 tasks of the application (DAG); v_1 and v_2 are mapped to N_1, and v_3 is mapped to N_2; the initial application security level between tasks v_2 and v_3 is level 3, L_3, and the environment security level between nodes N_1 and N_2 is L_5. After executing the tasks v_1 and v_2, node N_1 has to encrypt the transmitted data according to the specific security requirements. Node N_2 needs to firstly decrypt the received data and then execute task v_3. The security level L_5 will be selected for both the encryption and decryption.

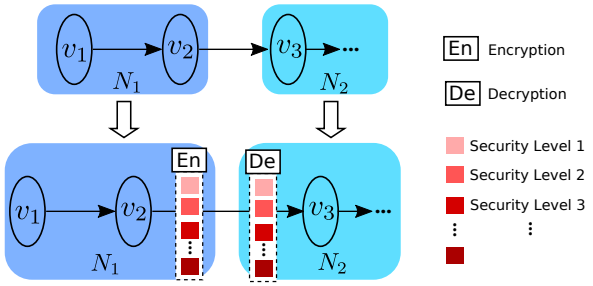

Fig. 1. DAG graph for application with critical message security.

C. Cost Functions

The main activities of one sensor node are normal task processing, data encryption and decryption, transmitting, receiving and sleeping. As the sleeping power is typically very small [9], it is neglected in this work[1].

The execution time of node N_i for executing task v is $t_i(v) = w(v)/f_i$, where $w(v)$ stands for the computation workload (the number of CPU clock cycles) of task v and f_i is the processing speed of node N_i. The processing cost is therefore formulated as:

$$e_{i\_p}(v) = P_i\, t_i(v) \tag{1}$$

where P_i represents the average processing power of node N_i.

The execution time of data encryption and decryption are considered the same in this work. According to [4], the execution time for encrypting/ decrypting H bits of data at the selected security level L_j are:

$$t^j_{en/de} = I + \theta(L_j)H \tag{2}$$

where I is the executing time for pre-/post-whitening on the encryption and decryption; $\theta(L_j)$ is the cryptographic computation time for one bit of data at the jth security level. The corresponding encryption and decryption cost of N_i are:

$$E^j_{i\_en} = E^j_{i\_de} = P_i t^j_{en/de} \tag{3}$$

The communication procedure of the sensor node includes both the overhead activities and data packets communication, as described in [10]. The corresponding energy cost of node N_i for transmitting and receiving H bits of data, E_{i_tx} and E_{i_rx}, can be formed by:

$$E_{i\_tx} = e_o + e_{tx}(d_i)H \quad \text{and} \quad E_{i\_rx} = e_o + e_{rx}H \tag{4}$$

where e_o is the overhead energy cost of the communication, d_i is the transmitting distance of node N_i, $e_{tx}(d_i)$ and e_{rx} are the energy spent for transmitting and receiving 1 bit of data, respectively. The communication time for 1 wireless hop is:

$$t_{cmm} = t_o + H/BW \tag{5}$$

where t_o is the overhead duration of the communication and BW is the bandwidth of the wireless radio.

III. SATA Algorithm

This section firstly formulates the security aware task allocation problem and then presents the SATA algorithm.

A. Problem Modeling

The security aware task allocation for multihop wireless networks is to map the whole DAG to the network subject to the security requirements and the application execution $Deadline$. The objective of this work is to find the best mapping (task allocation solution) to maximize the network lifetime (NL). We define NL as the duration from the network starts until the first node runs out of battery, which has been widely used in current references, such as in [11], [12].

Based on GA, one complete task allocation solution with the security level selection is represented by a chromosome, $\boldsymbol{C}$. By applying the m-th chromosome, $\boldsymbol{C}_m$, the corresponding NL_m can be formulated by:

$$NL_m = min\{\frac{Bat_1}{E_1}, \cdots, \frac{Bat_i}{E_i}, \cdots \frac{Bat_n}{E_n}\} \tag{6}$$

where Bat_i is the battery energy of node N_i; and E_i denotes the total energy cost of node N_i in one scheduling round of the given application including the cost of receiving, decrypting, processing, encrypting and transmitting. Thus, SATA aims to find the best chromosome $\boldsymbol{C}_m$ corresponding to the maximum NL under the security constraints and user defined $Deadline$.

B. SATA

SATA algorithm consists of three main components: 1) model the security aware task allocation solutions as genetic genes, i.e., chromosomes; 2) design fitness function; 3) evolution process including inheritance, crossover and mutate to produce new generations. The evolution process of SATA is repetitively executed until the user-defined maximum iteration number. According to the nature law, the good chromosomes will be accumulated and improved while the bad ones will be eliminated. SATA will select the best chromosome in the last generation as the final security aware task allocation solution.

1) Genetic formation of the solution: Consider that the application has K tasks, the possible mapping of the tasks to the sensor nodes is modeled by a chromosome, $\boldsymbol{C}$, i.e., a 1-by-K vector. The orders of the elements represents the tasks. The K elements of $\boldsymbol{C}$ are the randomly generated node IDs from 1 to n. The initial energy matrix of the sensor nodes is modeled by a 7-by-K matrix. The rows in order stand for the tasks in order, node IDs, the receiving, decryption, processing, encryption

[1]SATA can be directly extended for covering the sleeping cost.

and transmitting energy cost. Considering the multiple wireless hops, the energy matrix needs to be extended by adding the cost of routing nodes. The total schedule time for completing the given application is therefore made up of the processing time of each task, the encryption and decryption time, and the communication time. Fig. 2 illustrates one example. The chromosome $\boldsymbol{C} = [N_1, N_1, N_3, N_6, \cdots]$ represents: tasks v_1 and v_2 are assigned to node N_1; tasks v_3 and v_4 are assigned to nodes N_3 and N_6, respectively. As node N_2 is the routing node for the connection between nodes N_1 and N_3, the energy matrix, $\boldsymbol{E}$, is:

$$\boldsymbol{E} = \begin{bmatrix} v_1 & v_2 & 0 & v_3 & v_6 & \cdots \\ N_1 & N_1 & N_2 & N_3 & N_6 & \cdots \\ 0 & 0 & E_{2\_rx} & E_{3\_rx} & E_{6\_rx} & \cdots \\ 0 & 0 & E^j_{2\_de} & E^j_{3\_de} & E^j_{6\_de} & \cdots \\ e_{1\_p}(v_1) & e_{1\_p}(v_2) & 0 & e_{3\_p}(v_3) & e_{6\_p}(v_4) & \cdots \\ 0 & E^j_{1\_en} & E^j_{2\_en} & E^j_{3\_en} & E^j_{6\_en} & \cdots \\ 0 & E_{1\_tx} & E_{2\_tx} & E_{3\_tx} & E_{6\_tx} & \cdots \end{bmatrix}$$

The application scheduling time, ST, can be calculated by:

$$\begin{aligned} ST =& t_1(v_1) + t_1(v_2) + t^j_{en} + t_{cmm} + t^j_{de} + t_3(v_3) + t^j_{en} + t_{cmm} \\ &+ t^j_{de} + t_6(v_4) + t^j_{en} + t_{cmm} + \cdots \end{aligned} \tag{7}$$

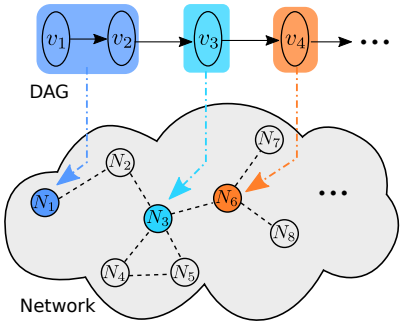

Fig. 2. A chromosome format of one possible task allocation solution, $\boldsymbol{C} = [N_1, N_1, N_3, N_6, \cdots]$.

Based on the energy matrix $\boldsymbol{E}$ and Eqs. (6) and (7), the network lifetime NL_m and ST_m by applying chromosome $\boldsymbol{C}_m$ can be easily calculated.

2) Fitness function: Since the security requirement has been considered in the generated chromosomes, the fitness function should further consider NL and ST simultaneously. The normalized fitness function is a widely used candidate, such as in [12], [13]. This work chooses the one used in [13] as shown in the following:

$$fit_m = \frac{NL_m}{max\{NL_{1,\ldots}, NL_M\}} - \alpha \frac{ST_m}{max\{ST_{1,\ldots}, ST_M\}} \tag{8}$$

where M is the population number of one generation; $\alpha = 1$ when $SL_m > Deadline$, otherwise $\alpha = 0$. A chromosome is better when it has a higher fitness value.

3) Evolution process of SATA: The evolution process of SATA consists of inheritance, crossover and mutation of the chromosomes. Firstly, the inheritance directly saves the η ratio of the best chromosomes into the next generation. In other words, ηM chromosomes with the highest fitness values of Eq. (8) will be inherited. Secondly, the rest $(1-\eta)M$ will be paired based on the roulette wheel scheme [14] before the crossover. For each pair of chromosomes, single point crossover method is used to produce the offspring. A crossover point will be randomly generated between 1 to K, the two paired chromosomes switch over after the crossover point. Thirdly, the mutation is employed by SATA to enhance the genetic diversities (i.e., prevent chromosomes from being too similar), thereby reducing the chance of getting stuck in local optimum. Each chromosome in the generation has a probability of μ to be replaced by a newly generated chromosome.

Based on the genetic formation and the fitness function as presented in Sections III-B1 and III-B2, SATA repeatedly executes its evolution process as presented in Section III-B3 until the user defined maximum iteration number. The chromosome with the best fitness value in the last generation will be chosen as the final security aware task allocation solution.

IV. Evaluation

This section evaluates the proposed SATA algorithm using extensive simulation results. We demonstrate the superiority of SATA by comparing with the greedy algorithm and ITAS[2] algorithm proposed in [12]. For fair comparison, this work follows the experimental setup in [12]: The multihop wireless network is randomly generated with one sink node at the center and n randomly distributed sensor nodes. The energy parameters are taken from [12]. The security requirements for the tasks and the sensor nodes are randomly generated from security levels 1 to 5. The related security parameters are taken from [4]. The DAG is randomly generated, in which the computing workload of each task is within the range of $[100, 500]$ KCCs (kilo clock cycles) and the communication data on each edge is in the range of $[100, 1000]$ bits. The $Deadline$ of DAG is 60 ms. The chromosome population in one generation is $M = 40$. The inheritance ratio and mutation probability are $\eta = 20\%$ and $\mu = 1\%$, respectively.

The performance on network lifetime, algorithm runtime and application scheduling time are investigated by changing a) the number of nodes, n, b) the number of tasks, K. Only one parameter is changed in each simulation. The reported results correspond to the average values of 200 test instances.

The first set of simulations is conducted to investigate the algorithm performance by changing the number of sensor nodes, n. As shown in Fig. 3a, SATA has longer network lifetime than the greedy and ITAS [12]. As n changes from 5 to 40, the network lifetime firstly dramatically increases and then keeps stable. This is due to the fact that a higher number of nodes can achieve a more balanced sharing of the tasks. When n is larger than the number of tasks, the extra sensor nodes are not able to share any of the tasks. As a result, the network lifetime cannot be extended further. When looking at the algorithm runtime in Fig. 3b, SATA requires similar execution time as ITAS [12] and both of them are longer than the greedy algorithm. They need more time for the algorithm execution when n increases. Since the increased sensor nodes bring more multihop wireless communication, the calculations

[2]ITAS in [12] can provide the optimal solution based on GA without the consideration of security concern. After ITAS gets the final results, this work work adds the extra security requirement to the results.

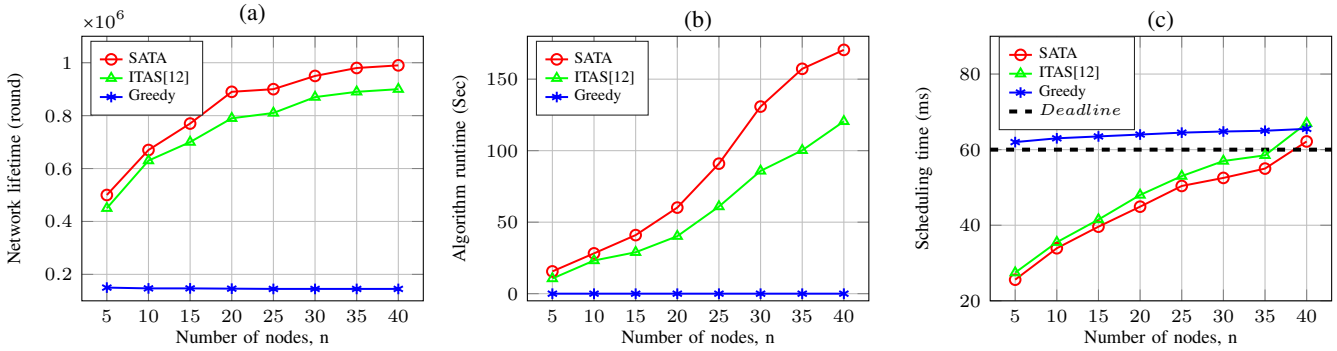

Fig. 3. The impact of number of sensor nodes on (a) network lifetime, (b) algorithm runtime and (c) application scheduling time (10 tasks in the DAG).

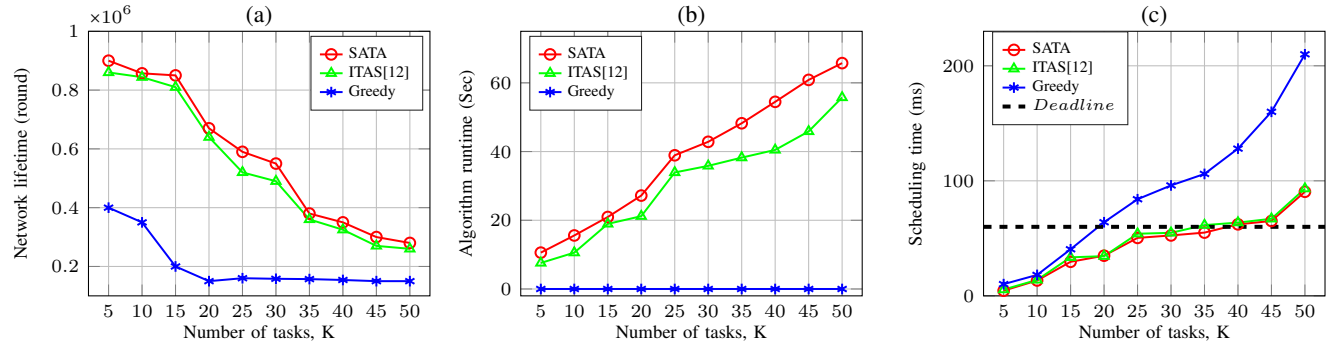

Fig. 4. The impact of number of tasks on (a) network lifetime, (b) algorithm runtime and (c) application scheduling time (10 sensor nodes in the network).

of the energy matrix $\boldsymbol{E}$ and Eq. (7) require longer time. Due to the same reason, the application scheduling time increases as n changes from 5 to 40 as shown in Fig. 3c. Nevertheless, SATA can still satisfy the predefined $Deadline$.

Moreover, we further vary the number of the tasks, K, in the DAG to estimate the performance of SATA. It can be seen from Fig. 4a that SATA performs the best as expected. When K increases, the network lifetime of SATA decreases. It can be explained by two reasons: firstly the computing workload for the whole network increases as K changes from 5 to 50; secondly, the energy cost for multihop wireless communication is also increased as K increases. Moreover, as the chromosome formulation becomes more complex as K increases, SATA requires longer time for the algorithm execution as shown in Fig. 4b. Since the application scheduling time is the completing time of the total tasks, it is obviously increased when K becomes larger as demonstrated in Fig. 4c.

Based on the above simulation results, SATA can efficiently extend the network lifetime while guaranteeing the security requirements for small-to-medium multihop wireless networks.

V. Conclusion

This work simultaneously considers both the energy efficiency and security requirements for the emerging multihop wireless networks. An intelligent **s**ecurity **a**ware **t**ask **a**llocation algorithm (SATA) is proposed based on genetic algorithm. SATA models the security aware task allocation solution by chromosomes, and uses a hybrid fitness function for ranking the solutions. By repeatedly executing the evolution processes of SATA, i.e., inheritance, crossover and mutation, the final optimal solution can be achieved. The simulation results illustrate significant improvements in different test scenarios comparing with other algorithms.

References

[1] M. Kocakulak and I. Butun, "An overview of wireless sensor networks towards internet of things," in *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, Jan 2017, pp. 1–6.

[2] W. Yu, Y. Huang, and A. Garcia-Ortiz, "Modeling optimal dynamic scheduling for energy-aware workload distribution in wireless sensor networks," in *2016 International Conference on Distributed Computing in Sensor Systems (DCOSS)*, May 2016, pp. 116–118.

[3] W. Yu, Y. Huang, and A. Garcia-Ortiz, "Optimal task allocation algorithms for energy constrained multihop wireless networks," *IEEE Sensors Journal*, pp. 1–1, 2019.

[4] W. Jiang, P. Pop, and K. Jiang, "Design optimization for security- and safety-critical distributed real-time applications," *Microprocess. Microsyst.*, vol. 52, no. C, p. 401415, Jul. 2017.

[5] H. Nam and R. Lysecky, "Mixed cryptography constrained optimization for heterogeneous, multicore, and distributed embedded systems," *Computers*, vol. 7, no. 2, 1 2018.

[6] M. Mitchell, *An Introduction to Genetic Algorithms*. Cambridge, MA, USA: MIT Press, 1998.

[7] S. S. Chiang, C. H. Huang, and K. C. Chang, "A minimum hop routing protocol for home security systems using wireless sensor networks," *IEEE Trans. on Cons. Elec.*, vol. 53, no. 4, pp. 1483–1489, Nov. 2007.

[8] Y. Sahni, J. Cao, and L. Yang, "Data-aware task allocation for achieving low latency in collaborative edge computing," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3512–3524, April 2019.

[9] Texas-instruments, *CC2538 Datasheet*, Chipcon Products from Texas Instruments, USA: Texas Instruments, Copyright, 2013.

[10] Y. Huang *et al*, "Accurate energy-aware workload distribution for wireless sensor networks using a detailed communication energy cost model," *J. of Low Power Elec.*, vol. 10, no. 2, pp. 183–193, June 2014.

[11] W. Yu, Y. Huang, and A. Garcia-Ortiz, "An altruistic compression-scheduling scheme for cluster-based wireless sensor networks," in *2015 12th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, June 2015, pp. 73–81.

[12] Y. Jin, J. Jin, A. Gluhak, K. Moessner, and M. Palaniswami, "An intelligent task allocation scheme for multihop wireless networks," *IEEE Trans. on Para. and Dist. Syst.*, vol. 23, no. 3, pp. 444–451, March 2012.

[13] A. G. Y. Jin, S. Vural and K. Moessner, "Dynamic task allocation in multi-hop multimedia wireless sensor networks with low mobility," *Sensors*, vol. 13, no. 10, pp. 13 998–14 028, Oct. 2013.

[14] D. Goldberg, *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley Longman Publishing co., 1989.