

Simulated Annealing algorithm for Data Aggregation Trees in Wireless Sensor Networks

Ladan Darougaran¹, *Hossein Shahinzadeh², Mohammadreza Salavati³

¹Young Researchers Club, Tabriz Branch, Islamic Azad University, Tabriz, Iran,
Ladan_darougaran@Yahoo.com

²Young Researchers Club, Khomeini Shahr Branch, Islamic Azad University, Khomeini
Shahr, Iran, s.shahinzadeh@iaukhsh.ac.ir

³Khomeini Shahr Branch, Islamic Azad University, Khomeini Shahr, Iran,
Salavati@iaukhsh.ac.ir

Abstract. Wireless sensor networks look like mobile ad hoc networks based on many aspects, but protocols which are used for ad hoc networks, are not suitable for wireless sensor networks. In ad hoc networks, the main issue about designing of protocols is quality of service, so that in wireless sensor networks the main constraint in designing protocols is limited energy of sensors. In fact, protocols which minimize the power consumption in sensors are more considered in wireless sensor networks. One approach of reducing energy consumption in wireless sensor networks is to reduce the number of packages that are transmitted in network. The technique of collecting data that combines related data and prevent transmission of additional packages in network can be effective in the reducing of transmitted packages' number. According to this fact that information processing consumes less power than information transmitting, Data Aggregation has great importance and because of this fact this technique is used in many protocols [5]. One of the Data Aggregation techniques is to use Data Aggregation tree. But finding one optimum Data Aggregation tree to collect data in networks with one sink is a NP-hard problem. In the Data Aggregation technique, related information packages are combined in intermediate nodes and form one package. So the number of packages which are transmitted in network reduces and therefore, less energy will be consumed that at last results in improvement of longevity of network. Heuristic methods are used in order to solve the NP-hard problem that one of these optimization methods is to solve Simulated Annealing problems. In fact, SA is derived from melting process and re-cooling of materials, so it is called Simulated Annealing. SA does not present the best result necessarily. But SA sake one good answer that can also be optimum [6]. In this article, we will propose new method in order to build data collection tree in wireless sensor networks by using Simulated Annealing algorithm and we will evaluate its efficiency.

Keywords: Data aggregation, Wireless sensor networks, energy efficiency, Simulated Annealing algorithm

1. Introduction

Recent advances in wireless communications and electronics have enabled the development of low-cost, low-power, multifunctional sensor nodes that are small in size and communicate in short distances. Wireless sensor networks are becoming a rapidly developing area in both research and application. Although it was originally driven by military applications, wireless sensor networks are being investigated and applied in many different civilian applications. For example, the sensor network has been applied for vehicle tracking system, habitat monitoring, forest surveillance, earth quake observation, biomedical or health care applications and building surveillance. A wireless sensor network consists of a number of sensor nodes scattered in the region of interest in order to acquire some physical data. The sensor node should have the ability of sensing, processing and communicating

[1]. A wireless sensor network operates in an unattended environment, with limited computational and sensing capabilities, and capable of sensing, computing and communicating wirelessly. In order to effectively utilize wireless sensor nodes, we need to minimize energy consumptions in the design of sensor network protocols and algorithms. A large number of sensor nodes have to be networked together. Direct transmission from any specified node to a distant sink node is not used since sensor nodes that are farther away from the sink node will drain their power sources much faster than those nodes that are closer to the sink node. On the other hand, a minimum energy multi-hop routing scheme will rapidly drain the energy resources of the nodes, since these nodes are engaged in the forwarding of a large number of data messages (on behalf of other nodes) to the sink node. Thus, one solution is to use multi-hop communication with in-network aggregation of correlated data [7]. The application of an aggregation approach helps to reduce the amount of information that needs to be transmitted by performing data fusion at the aggregate points before forwarding the data to the end user [8]. The Simulated Annealing optimization is one of the most successfully proven swarm intelligence. It has been successfully applied in many difficult discrete optimization problems such as the traveling salesman problem [9], scheduling, vehicle routing, etc., as well as routing in wireless networks [10]. In this paper, we proposed a data aggregation mechanism in wireless sensor networks using Simulated Annealing algorithm. The remainder of this paper is organized as follows. Section and Section II describes Simulated Annealing algorithm and our method and also Section III shows simulation results and Section IV Conclusion the paper.

2. An overview of simulated annealing algorithm

Simulated Annealing [13] is technique used to find the optimal solution in the solution space for an optimization problem. The Simulated Annealing algorithm starts with an initial solution S_0 at initial Temperature T_0 and $Time=0$ in lines 2-4. An initial solution is most often generated randomly but a heuristic can be used. In the annealing process, new configuration of the current solution is produced and accepted if its cost is better than the cost of the current configuration or within probability as temperature decreases. In the do while loop in lines 5-12, Metropolis function invokes Neighbor function to produce the new configurations using a copy of the initial solution in line 6. The cooling rate which controls the temperature is α ($0 < \alpha < 1$) in line 7. In line 8, M represents the number of configurations that will be produced at temperature T ($M > 1$). This process continues until the Time exceeds the maximum time. In line 13, the Best solution found is returned.

The structure of the Metropolis algorithm:

```
1 begin
2   Time=0;
3   T=T0;
4   S=S0;
5   Repeat
6     Metropolis(S,T,M);
7     T= $\alpha$ *T;
8     M= $\beta$  *M;
10    Time=Time +M;
12  Until (Time> Max time)
13  Return Best S
14 end
```

The structure of the Metropolis algorithm:

```
1 begin
2   Repeat
3     NewS=Neighbor(S);
4     NewCost=Cost(NewS);
5      $\Delta$ Cost=NewCost-Cost(S);
```

```

6         if( $\Delta\text{Cost}<0$  OR  $\text{RANDOM}<\text{exp}(-\Delta\text{Cost}/T)$ )
7             NewS=S;
8             M=M-1;
9     Until(M=0)
10 end
    
```

The do while loop in lines 2-9 represents the process of producing and evaluating the new configuration by invoking the Neighbor function(line 3) and Cost function (line 4) respectively. There are four operators to produce the new solution or configuration. Move operator which moves an actor selected randomly to a new location. The Swap operator which swaps the location of two randomly selected actors. Then the Add operator, to add new actor selected from the actors-library at random empty position. Finally, the Delete operator to delete an actor selected randomly. In line 4, the Cost function evaluates the solution and returns the total cost of the actors in the solution. In line 6, the new solution is accepted if its cost is less than the cost of the current solution or if the probability returned by $\text{exp}(-\Delta\text{Cost}/T)$ is greater than a random value ($0<\text{RANDOM}<1$). The probability of accepting a worse solution increases when the temperature is high and decreases when the temperature is low. Line 9 shows that the process will end when M equals zero.

3. Simulated Annealing algorithm and our method

3.1. Demonstration of Response Structure

Response structure demonstrates one point of a feasible space of a problem, so that the way of its demonstration in every super-innovative approach is important. The response structure we have considered is as following:

1	2	3	4	98	99	100
5	6	21	100	2	54	43

Figure 1. Demonstrates response structure

Figure 1 demonstrates response structure, so that indices of array indicates ID nodes and content of array represents parent of each node, and ID parents of everyone is specified; for example, parent of No.4 node is No.100 node. Another one is No.1 which we have considered it as "sink", or in another terms, as a root of tree.

3.2. Selection of Primary Response

Selection of a proper primary response highly increases convergence speed toward global optimal response. To increase efficiency of algorithm, primary response is controlled, whether it is acceptable or not. To be acceptable means that it has the characteristics of a tree (connectedness and without circle). Figure 2 shows pseudo-code control of a tree.

```

Function Control (Solution)
Flag = False
While flag
    Create (Tree);
    Control (Loop & Connectivity)
    If Control = OK then flag = True
END
    
```

Figure 2. pseudo-code control of a tree

3.3. Mechanism of Creating Neighbor Response

To survey in a feasible space, we need to produce another response by changing the current one, which is referred to as "neighbor response". The mechanism which we have chosen for producing neighbor [response] is that two nodes are selected randomly and parents of them is exchanged; feasibility of response is then studied. There is no guarantee that the produced neighbor is certainly a tree. If the obtained response is not confirmed, it is eliminated and another one is produced. Figure 3 shows Mechanism of Creating Neighbor Response.

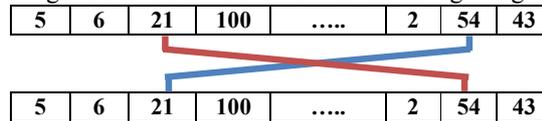


Figure 3. Mechanism of Creating Neighbor Response

3.4. Selection of Primary Temperature

Determination of primary temperature value of a system (T_0) has obvious and direct contribution to acceptance or refusal of responses, because in high primary temperature state, system energy is also high, and it is a desirable state to find best path to reduce temperature in attaining a stable state (most stable energy level). When primary temperature is selected to be low, it is unlikely that worse response became acceptable and the system may remain in optimal local state.

3.5. Fitness Function

We have considered fitness of any response as energy consumption of every tree. It is also calculated on the basis of energy principle (Heinzelman et al, 2005).

3.6. Mechanism of Reduction in Temperature

It is used formula (1) for measure of reduction in temperature and moving toward system cooling, where α represents coefficient of reduction in temperature, it is constant of <1 . Its values usually considered 0.8, 0.9, and 0.95. Parameter of n demonstrates the system's number of reductions in temperature level. It is also used as measure to halt SA algorithm.

$$T_i = \alpha T_{i-1} \quad \forall i = 1, \dots, n \quad (1)$$

3.7. Markov Chain Length

One of the important parameters to determine the quality of resulting responses from SA is the number of searched points in the space of problem response in any temperature. We refer to the number of these responses as Markov Chain Length. It is necessary to determine this parameter in order to ensure that close search has been performed to all possible responses. The simplest suggestion to determine the value of Markov Chain Length in SA is to choose a constant value, which is independent of primary temperature, which according to (while), it must be close to the value of the problem as far as possible.

3.8. Standards to Halt SA Algorithm

In SA algorithm some standards are:

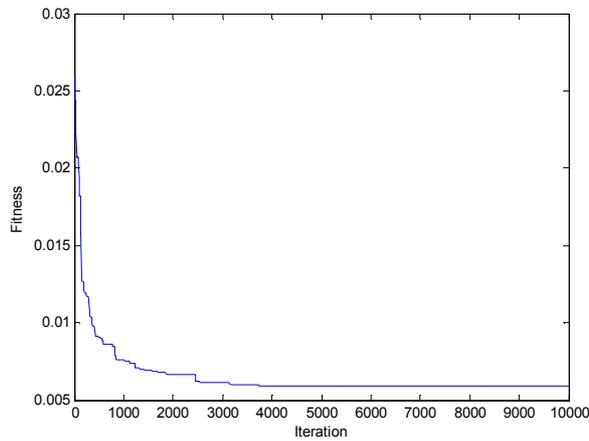
- The number of reductions in temperature level.
- The successive number in cooling process, which is not resulted in improvements in target function.
- Accessing to desirable lower temperature level.

4. Simulation results

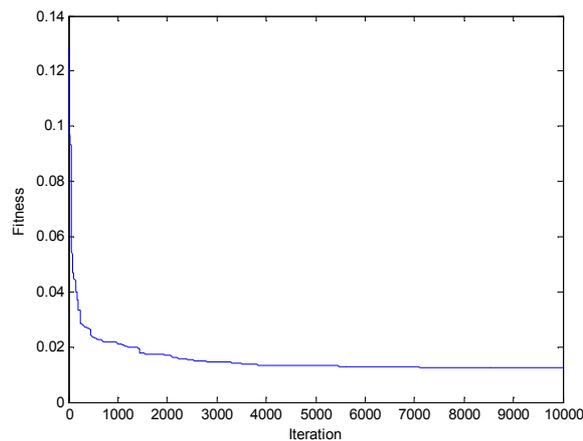
For simulation experiments, we assumed that there are sensor nodes distributed randomly in a $100m \times 100m$ square region. All nodes have the same transmission range. There is a single sink node located at coordinates (10, 10) of the wireless sensor networks, which receives the data of all source nodes for all the simulations.

The first set of experiments is carried out to investigate the total energy consumption with $N=100$, $N=200$ and $N=500$ fig4 (a-c) shows the results. The axis X shows the number of runs and the axis Y shows fitness function. Fitness function chooses path which consumes low energy. It's clear that SA is getting convergence to improvement tree with lowest consumption energy because of this all diagrams are descent.

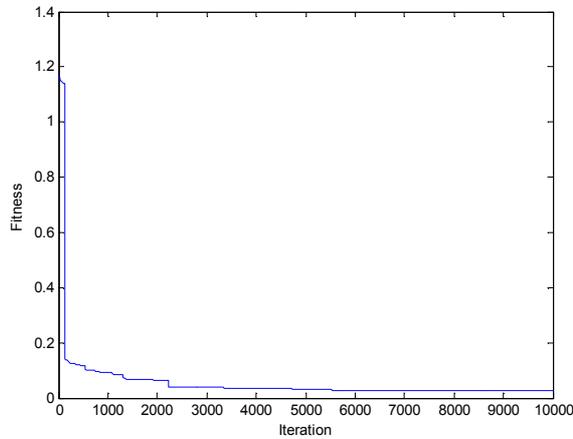
For instance fig 4-c for $N=500$ first solution has been offered for SA which consumes energy about 0.19J for sending data aggregation result to sink but SA algorithm gives us the best solution in 10000th run. Latest solution consumes energy about 0.069J. It means best data aggregation path.



(a)



(b)



(c)

Figure 4. Fitness function calculates total energy consumption (a)N=100,(b)N=200,(c)N=500

Fig 5 is temperature reduction. This event is very importance in SA algorithm. We considered $T_0=20000$ and $\alpha=0.98$. If temperature reduction happens very slowly, probability of best solution will increase. If temperature reduction happens very fast, SA will tarp in local solution.

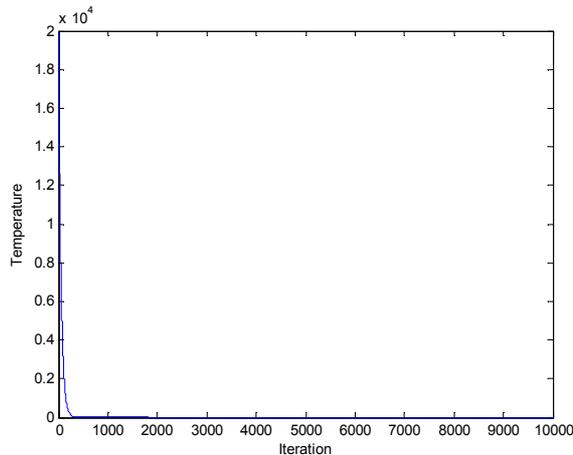


Figure 5. Temperature reduction

We assume in simulation that coordinate of sense area is distinct. There are nodes which located in this area are as source nodes. They are leaf of data aggregation tree. We start to arrange data aggregation tree in this location because aggregation result will be valid. If tree has leafs outside of sense area, result of this tree will be invalid because invalid data gathered our data. Because of that we prevent this event in our simulation by omitted invalid leafs. Middle nodes are as aggregator nodes and they do data aggregation and send result to upper level. We consider that sink is root of our tree. Aggregation result is sent to root. Fig 6-a shows invalid tree and fig 6-b shows valid tree.

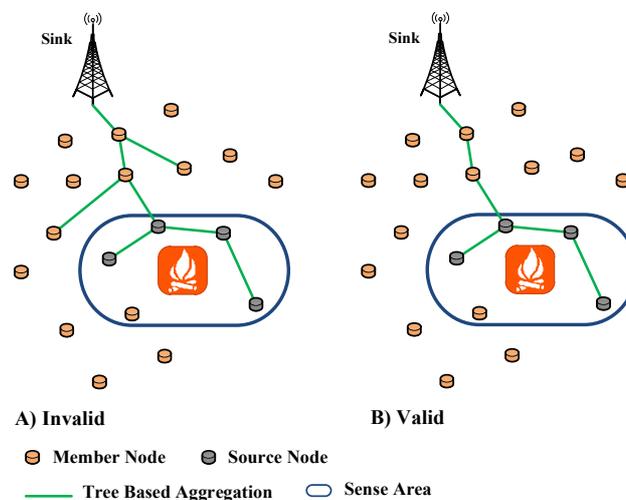


Figure 6. Data aggregation tree

5. Conclusion

One of the existing problems in data aggregation algorithms is that in these algorithms the shortest path between starting and ending node is always selected for transmission between the two nodes, which makes selected energy nodes to be evacuated rapidly. Particularly this problem shows itself when our network has great scale and the rate of data being transferred from a distinct region is relatively high, which is quite possible in sensor networks. The other negative point which worsens this problem is that if a path is evacuated, mostly the other shortest path, which typically is near this one, is selected; especially when the distance between starting and ending point node is farther, it can gradually cause different parts of network to be separated. Therefore, we need a method able to distribute equitably traffic of transmitted data between starting and ending points among path nodes. Performed simulations indicate that we can access such a tree using Simulated Annealing algorithm.

6. References

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci., "Wireless sensor networks: a survey", *Journal of Computer Networks*, Vol. 38, March 2002, pp.393-422.
- [2] R. H. Katz, J. M. Kahn and K. S. J. Pister, "Mobile Networking for Smart Dust", *Proc. of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, Seattle, USA, Aug. 1999, pp. 350-355.
- [3] A. Manjeshwar and D. P. Agarwal, "TEEN: A Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks", *Proc. of the IEEE IPDPS*, San Francisco, USA, Apr. 2001, pp 23-26.
- [4] J. M. Rabaey, M. J. Ammer, J. L. da Silva, D. Patel and S. Roundry, "PicoRadio supports ad hoc ultra-low power wireless networking", *IEEE Computer Magazine*, Vol. 33, Jul. 2000, pp. 42-48.
- [5] B. Krishnamachari, D. Estrin, S. Wicker, "Modeling Data Centric Routing in Wireless Sensor Networks", *Proc. of the IEEE INFOCOM*, New York, USA, Jun. 2002.
- [6] Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, 1953. Equation of State Calculation by Fast Computing Machines. *J. of Chem. Phys.*, 21, 1087-1091.
- [7] H. Luo, Y. Liu, Das SK. Routing correlated data with fusion cost in wireless sensor networks. *IEEE Trans. Mobile Comput.* 2006;5(11):1620–32.

- [8] D. Li, J. Cao, M. Liu, Y. Zheng. Construction of optimal data aggregation trees for wireless sensor networks. In: International Conference on Computer Communications and Networks (ICCCN), 2006.
- [9] M. Dorigo, LM. Gambardella Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. Evol. Comput. 1997;1(1):53–66.
- [10] G. Chen, T-D. Guo, W-G. Yang, Zhao T. An improved ant-based routing protocol in wireless sensor networks. In: International Conference on Collaborative Computing: Networking, Applications and Worksharing(CollaborateCom), 2006.
- [11] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, F. Silva Directed diffusion for wireless sensor networking. IEEE/ACM Trans. Networking 2003;11(1):2–16.
- [12] R. Misra, C. Mandal. Ant-aggregation: ant colony algorithm for optimal data aggregation in wireless sensor networks. In: International Conference on Wireless and Optical Communications Networks, 2006.
- [13] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, “Optimization by simulated annealing,” Journal of Science, vol. 220, no.4598, May 1983 pp.671-680