

A Free Market Economy Model for Resource Management in Wireless Sensor Networks

Byeongwook Bae¹, Junyoung Park², Sunggu Lee¹

¹Department of Electrical Engineering, Pohang University of Science and Technology, Pohang, Korea

²System LSI Division, Samsung Electronics Co., Ltd., Hwasung, Korea

Email: lidwoogi@postech.ac.kr, zenojur@gmail.com, slee@postech.ac.kr

Received 15 May 2015; accepted 26 June 2015; published 29 June 2015

Copyright © 2015 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

This paper presents a free market economy model that can be used to facilitate fully distributed autonomous control of resources in massive heterogeneous wireless sensor networks (WSNs). In the future, it is expected that WSNs will exist as part of the global Internet of Things (IoT), and different WSNs can work together in a massive network of heterogeneous WSNs in order to solve common problems. Control of valuable processing, sensing and communication resources, determining which nodes will remain awake during specific time periods in order to provide sensing services, and determining which nodes will forward other nodes' packets are difficult problems that must be dealt with. It is proposed that just as the free market economy model enables the global human society to function reasonably well when individuals simply attempt to trade money and services in order to maximize their individual profits, and a similar model and mechanism should enable a massive network of heterogeneous WSNs to function well in a fully distributed autonomous manner. The main contributions of this paper are the introduction of the free market economy model for use with WSNs, the formal definition of a maximum profit price problem for multihop packet relaying, and the proposal of a distributed genetic algorithm for the solution of the maximum profit price problem. Simulation results show that the proposed distributed solution produces results that are 70% - 80% similar to a pareto optimal solution for this problem.

Keywords

Distributed Control, WSN Resources, Free Market Economy, Auction, Autonomous Control

1. Introduction

With the widespread use of mobile devices supporting communication, many modern day living areas are sur-

rounded by various wireless and wired networks. The proliferation of these communication-capable devices creates the Internet of Things (IoT), in which nodes with unique IDs construct a network across diverse platforms and share data for their own purposes [1]. A central component of an IoT is a wireless sensor network (WSN) [2], of which various different types, composed of a variety of sensors, can be used to provide various different services to the users of the IoT. In the future, just as the Internet has spread across the globe and has become ubiquitous, the IoT is expected to grow in size to span the globe and become ubiquitous. This future IoT is expected to include many different wireless sensor networks (WSNs), composed of various different types of sensor nodes with widely varying sizes and capabilities.

In such massive heterogeneous WSNs, it will become extremely difficult to provide effective control of the available resources (sensors, processors, wireless radio modules, batteries, etc.) such that the system works well as a whole while ensuring long average lifetimes and high performance with low overall cost. As the number of WSN nodes increases to massive proportions, the only viable type of solution will be a fully distributed and autonomous control method.

Many of the “tasks” that a sensor node performs can be considered as “services” that benefit other nodes while incurring a “cost” to the node performing the service. For example, most of the area in a physical region that needs to be continuously monitored is covered by multiple sensors in order to provide extended sensing lifetimes by letting sets of nodes alternately enter sleep mode and to compensate for faulty nodes or external sabotage attempts. Thus, a node that stays active and provides processing and/or sensing services for a time period T incurs a “cost” (since valuable battery capacity must be utilized during T) while providing a “service” to neighboring nodes by permitting them to enter sleep mode during T . As another example, a node that relays packets between two other nodes in the network provides a “service” to those two other nodes while incurring a heavy “cost” since wireless radio transmission requires a high level of energy usage [3]. Clearly, it will benefit a node to not provide such sensing or packet relaying services. However, if all nodes behaved in such a selfish manner, then the WSN would become useless as no sensing or multihop packet deliveries would occur. Thus, for these types of distributed resource control problems, it is proposed that a free market economy model be used to provide fair distribution and use of resources and services in a fully distributed autonomous manner.

2. Free Market Economy Model

A free market economy model, based on an auction system, is proposed as a compensation scheme for the use of valuable sensor node resources for activities such as network processing, sensing and multihop packet relaying that can benefit other nodes in a massive heterogeneous WSN. For a given sensor node v , this compensation can take the form of credits that can then be used by v to request similar services from neighboring nodes. It can also take the form of actual money if a service provider uses such services to save costs by not having to introduce new sensor nodes to provide additional capabilities or by not building new wireless radio towers in sparsely populated areas. In this paper, “money” will be used to refer to both types of compensation instruments.

The target massive heterogeneous WSN can be modeled by an undirected graph $G = (V, E)$, where V and E are the sets of nodes (which could consist of sensor nodes, wireless access points, and gateway nodes) and wireless communication links, respectively, in the WSN. The nodes in V are time-synchronized using a mechanism such as that described in [4]. Time is partitioned into frames, but frame beacons to announce the beginnings of frames are not necessary since each node can determine the start time of a frame individually based on its globally time synchronized clock—the requisite time synchronization mechanism should operate in a completely distributed autonomous manner, such as in the method proposed in [4].

During each time frame, each node $v \in V$ determines prices for each service that it plans to offer and announces these prices to its neighboring nodes. The node v can also request services from neighboring nodes or go to sleep to conserve energy. However, for each service that node v requests, it must have the money available to pay for that service. If the node runs out of money, then it must set desirable service prices and wait to receive money from neighboring nodes for services rendered. If the node runs out of energy during this process, then it simply dies until its battery can be replaced. Thus, for each node $v \in V$, prices and requests for services should be determined in a manner that ensures the maximum “profit” for itself. Then, when operated properly, the free market model should ensure the longevity of the WSN while also ensuring a fair and efficiently operating WSN. As a first step towards showing the utility of the proposed free market economy model, the following sections will focus on one particular application, multihop packet relaying.

3. Problem Definition and Related Work

One WSN application that can benefit from the proposed free market economy model is the problem of multihop packet relaying. **Figure 1** illustrates the basic problem being addressed. At the start of a time frame, each node $v \in V$ needs to establish a price $p(v)$ that node v will charge in order to relay a fixed-size packet of data through itself. There are also a set of packet communication requests R . Each request $r \in R$ includes a source node $s(r) \in V$ and a destination node $d(r) \in V$. If $s(r)$ has a direct wired or wireless communication link to $d(r)$, then there is no cost for communication request r . However, if such a direct connection does not exist, then a multihop path to $d(r)$ is used provided such a path exists. In order to use this multipath path, node $s(r)$ must pay each intermediate node that it uses the relay price charged by that node. For example, in **Figure 1**, if a node v_1 decides to use the path $P = \{(v_1, v_2), (v_2, v_3), (v_3, v_4)\}$ to send a packet to v_4 , $p(v_2)$ and $p(v_3)$ are paid to v_2 and v_3 respectively. If node $s(r)$ does not have sufficient money to pay all of the relay prices for a given path, then $s(r)$ cannot use that path. The prices of the packet relaying costs charged by each node are renewed during the beginning of each frame and the paths formed are valid until the end of each frame.

For each time frame, node $v \in V$ should set its price $p(v)$ in a manner that provides the most benefit for itself. A price $p(v)$ is established with the knowledge that other nodes will only use node v and pay the price $p(v)$ if the price is competitive with those charged on alternative routes for each communication request. For a given request $r \in R$, a distributed shortest-path (based on the sums of the prices of all intermediate nodes on a path) finding algorithm, such as a distributed version of Dijkstra's shortest path algorithm [5] or Ad-hoc On-demand Distance Vector routing (AODV), which floods the network with packets that store accumulated path information and then uses the information stored in return packets to find routing paths [6], needs to be used to determine the path that node $s(r)$ will select. Given that prices are established for each time frame and each node needs to establish pricing policies that ensure longevity and effective operation based on the specific needs of that node, each node $v \in V$ can establish conditions for its price $p(v)$ during a given time frame. For example, if its remaining energy level is extremely low or it has accumulated an extremely large amount of money, then $p(v)$ can be constrained to be greater than a very large constant. On the other hand, if v lacks sufficient money to request multihop packet deliveries that are urgently required, then it should utilize any available information about the prices expected to be established by neighboring nodes in order to specify that its price $p(v)$ should be less than the minimum of its neighbor's prices.

The *expected profit* $f(v)$ for a node $v \in V$ during a given time frame is defined to be the difference between the total reward and total cost for node v during that time frame, where the total reward is the sum of the prices paid by all requests $r \in R$ that are expected to utilize node v for multihop communication and the total cost is the sum of the prices paid by node v for all of its multihop communication requirements. Then the following optimization problem can be defined.

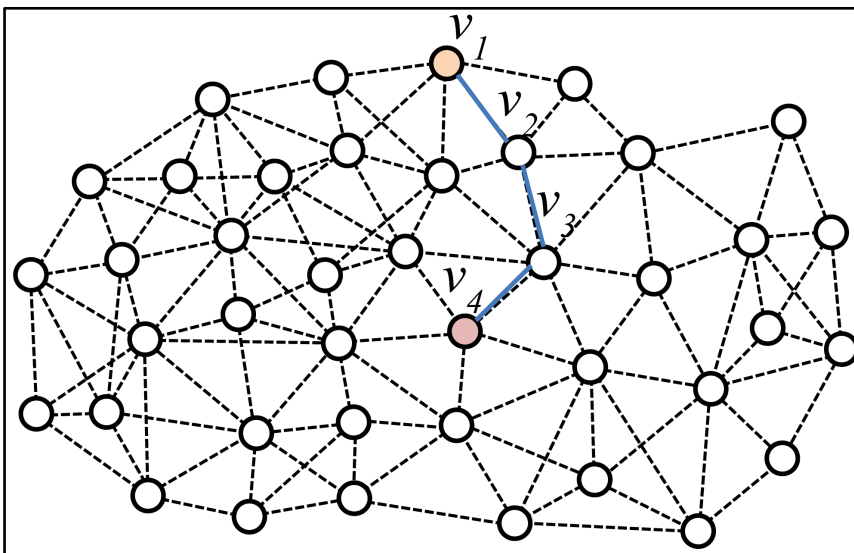


Figure 1. A multihop path from node v_1 to node v_4 .

PROBLEM: Max Profit Price (MPP)

INPUT: Graph $G = (V, E)$, set of requests R with source $s(r) \in V$ and destination $d(r) \in V$ for each $r \in R$. Each node $v \in V$ can optionally have a set of conditions that it sets on the price $p(v)$.

OUTPUT: Prices $p(v)^*$ for all nodes $v \in V$, where the price $p(v)^*$ is established such that the expected profit $f(v)$ is maximized given that all requests in R use the shortest paths, where the length of a path P is the sum of the prices on path P . The prices are established in a fixed random order (e.g., based on a hash function of the clock values of individual nodes in V) determined in a distributed autonomous manner, with prices carried over from the previous time frame until new prices can be determined for the current time frame.

In Problem MPP, which can be considered to be a variant of shortest path routing, since each node can have its own objective function, there may not exist a single optimal solution. For instance, the price that maximizes the profit of v_1 may minimize the profit of v_2 and vice versa. To address multiobjective optimization such as in Problem MPP, a *pareto optimal* solution, which is a solution in which no change can be made to improve one objective without degrading another objective, can be used [7]. A formal definition for *pareto optimal* can be based on Pareto's original work [8] as shown below.

Definition 1: A multidimensional point $x_m^* \in X$, is *pareto optimal* if there does not exist another point $x_j \in X$ such that there exists $i \in \{1, \dots, k\}$ with $F_i(x_j) > F_i(x_m^*)$, where $F_i(x_i)$, $i \in \{1, \dots, k\}$, is an objective function on $x_i \in X$.

Since multiobjective optimization is an NP-hard [9] problem, a heuristic approach is necessary. Thus, in this paper, the authors have investigated the use of evolutionary algorithms [10] [11] for the solution of Problem MPP. Previous works that have used multiobjective problem modeling to solve problems related to WSNs include approaches for sensor deployment, sensor coverage control, network lifetime enhancement, and multihop routing [12]-[14].

4. Proposed Algorithm

The authors propose an implementation of the non-dominated sorting and crowding distance methods [15] in order to search for a pareto optimal solution for Problem MPP. The proposed method is based on a genetic algorithm (GA), which is a popular type of evolutionary algorithm. A GA models possible solution instances as chromosomes and uses multiple iterations involving crossover and mutation, based on behavior observed in nature, in order to find successively better solutions.

A direct implementation of a GA solution for Problem MPP requires a centralized approach. However, a centralized algorithm is impractical for the type of massive heterogeneous WSN targeted in this paper. Thus, a distributed algorithm that imitates the operations of a GA is proposed as a practical solution to Problem MPP. However, a centralized GA solution will still be used, but simply as an "ideal" solution that is used to evaluate the quality of the proposed algorithm.

An "ideal" solution to the MPP problem, used for comparison purposes only, is shown below as **Algorithm CGA** (Centralized GA). In this algorithm, chromosome vectors in the population consist of the prices of nodes for packet relaying, in a decimal number format, and offspring are generated using arithmetic crossover. As a characteristic of multiobjective optimization, the algorithm has multiple fitness functions as many objective functions and parents for offspring are selected by non-dominated sorting based on the profit vector. **Algorithm CGA** is a GA that operates in a centralized manner. Because of the distributed nature and scale of the types of massive heterogeneous WSNs targeted, **Algorithm CGA** cannot be directly implemented for the target WSN.

The proposed algorithm, **Algorithm DGA** (Distributed GA), also shown below, is a distributed algorithm that imitates a GA and uses genetic operations such as selection and crossover within 1-hop subnetworks. Since operations involving setting prices, broadcasting price information, and finding shortest paths are required to incrementally produce solutions with higher cumulative profits, only one offspring is inserted into the population per round. Also, because the distributed algorithm cannot directly adopt a non-dominated sorting approach, the algorithm attempts to find solutions that are close to a pareto optimal solution by iteratively attempting to maximize one objective at a time. In order to do this, rather than faithfully implementing all aspects of a GA, **Algorithm DGA** simply uses a "GA-type" approach to maximize one objective at a time. Pseudocode descriptions of **Algorithms CGA** and **Algorithms DGA** are shown below.

5. Experimental Result

The performance of **Algorithm DGA** was evaluated using simulations. Since this is a completely new model for

Algorithm CGA. A genetic algorithm for Problem MPP, used for comparison purposes only.

$$C = \{c_0, c_1, \dots, c_{n-1}\}, F = \{f_0, f_1, \dots, f_{n-1}\}$$

// c_i = (price vector of nodes $v \in V$)
 // f_j = (profit vector of nodes $v \in V$), $|c_i| = |f_j| = |V| = N$

- 1) create initial population with $|C| = n$;
- 2) **for** ($i = 0$; $i < n$; $i++$) **begin**// initial phase
- 3) calculate profit vector $f_i \in F$ of $c_i \in C$;
- 4) **end for**
- 5) **while** (# of generation $< G$) **begin**// main phase
- 6) non-dominated sorting in C ;
- 7) select c_i and c_j using tournament selection;
- 8) arithmetic (average) crossover from c_i, c_j to p ;
- 9) mutate p to p_i ;
- 10) find the lowest non-dominated rank solution c_k from C ;
- 11) $C = C \cup \{p_i\} - \{c_k\}$;
- 12) **end while**

Algorithm DGA. The proposed distributed algorithm for Problem MPP.

$$C = \{c_0, c_1, \dots, c_{n-1}\}, F = \{f_0, f_1, \dots, f_{n-1}\}$$

// c_i = (price), f_j = (profit)

- For all** $v \in V$
- 1) create initial population $|C| = n$
 - 2) **for** ($i = 0$; $i < n$; $i++$) **begin**// initial phase
 - 3) $p(v) = c_i$ and notify $p(v)$;
 - 4) find shortest path if requests exist from v ;
 - 5) calculate profit $f_i \in F$;
 - 6) **end for**
 - 7) **while** (# of rounds $< G$) **begin**// main phase
 - 8) find index x with maximum profit from F ;
 - 9) find index y with minimum profit from F ;
 - 10) broadcast x to neighbor nodes;
 - 11) receive j_m from neighbor node m ;
 - 12) broadcast y to neighbor nodes;
 - 13) receive k_m from neighbor node m ;
 - 14) find the majority t_1 of j_m^y and x ;
 - 15) find the second majority t_2 of j_m^y and x ;
 - 16) $p = (c_{t_1} + c_{t_2})/2$; //generate offspring
 - 17) mutate p to p_i ;
 - 18) find shortest path if requests exist from v ;
 - 19) calculate profit f_i ;
 - 20) $r =$ (the majority of k_m^y and y)
 - 21) swap (c_0, c_i); swap (f_0, f_i);
 - 22) $c_0 = p_i$ and $f_0 = f_i$;
 - 23) **end while**

WSNs (based on all available open literature that the authors have been able to examine), other previous methods were not available for comparison. Thus, comparisons were made with a fixed price policy and a pareto optimal solution. *Angular similarity* to the pareto optimal solution, defined in [16], was used as the performance metric for profit vector comparisons.

The pareto optimal solution was obtained as the result of 500 generations of **Algorithm CGA**. **Algorithm DGA** also used 500 rounds in a distributed implementation. For each algorithm, the results, shown in **Figure 2**, were the averages of 10 different randomly connected networks. Overall, **Algorithm DGA** performed significantly better than a uniform fixed price policy. In the first few rounds, a few solutions produced by **Algorithm DGA** were below the results of the uniform fixed price policy. However, after 120 rounds, for almost every round, **Algorithm DGA** produced results that were closer to the pareto optimal solution (based on angular simi-

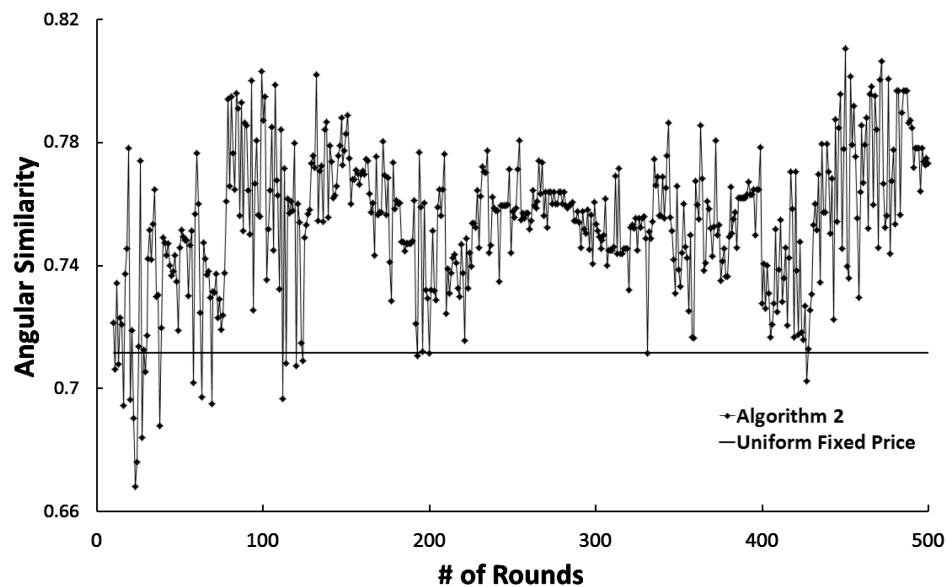


Figure 2. Angular similarity to the profit vector of a pareto optimal solution.

larity) than the uniform fixed price policy-this shows that **Algorithm DGA** derives better solutions as more rounds are used. As the number of rounds was increased, in certain cases, the similarity rapidly dropped before going back up because of the mutation nature of a GA-in a GA, mutations are used to avoid local minima solutions.

6. Conclusion

This paper has proposed the use of a free market economy model for the effective control of the resources in a massive heterogeneous WSN. The utility of this model is described in general times, and then a specific target application, packet relaying for multihop wireless networks, is examined. A maximum profit price problem is defined in a formal manner and a distributed algorithm solution, based on genetic algorithm principles, is proposed. The proposed algorithm is compared with a fixed price policy and a pareto optimal solution. Simulation results demonstrate that the proposed distributed algorithm performs similarly to the pareto optimal solution and performs successively better as more iteration rounds are used.

References

- [1] Gubbi, J., Buyya, R., Marusic, S. and Palaniswami, M. (2013) Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions. *Future Generation Computer Systems*, **29**, 1645-1660. <http://dx.doi.org/10.1016/j.future.2013.01.010>
- [2] Yick, J., Mukherjee, B. and Ghosal, D. (2008) Wireless Sensor Network Survey. *Computer Networks*, **52**, 2292-2330. <http://dx.doi.org/10.1016/j.comnet.2008.04.002>
- [3] Jang, U., Lee, S. and Yoo, S. (2012) Optimal Wake-Up Scheduling of Data Gathering Trees for Wireless Sensor Networks. *Journal of Parallel and Distributed Computing*, **72**, 536-546. <http://dx.doi.org/10.1016/j.jpdc.2012.01.008>
- [4] Marler, R.T. and Arora, J.S. (2004) Survey of Multi-Objective Optimization Methods for Engineering. *Structural and Multidisciplinary Optimization*, **26**, 369-395. <http://dx.doi.org/10.1007/s00158-003-0368-6>
- [5] Dijkstra, E.W. (1959) A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, **1**, 269-271. <http://dx.doi.org/10.1007/BF01386390>
- [6] Perkins, C.E. and Royer, E.M. (1999) Ad-Hoc On-Demand Distance Vector Routing. *2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, 25-26 February 1999, 90-100. <http://dx.doi.org/10.1109/mcsa.1999.749281>
- [7] Zitzler, E. and Thiele, L. (1999) Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation*, **3**, 257-271. <http://dx.doi.org/10.1109/4235.797969>

- [8] Pareto, V. (1906) *Manuale di Economia Politica*. Vol. 13, Societa Editrice.
- [9] Garey, M.R. and Johnson, D.S. (1979) *Computers and Intractability: A Guide to NP-Completeness*. Freeman Press.
- [10] Deb, K. (2001) *Multi-Objective Optimization Using Evolutionary Algorithms*. Vol. 16, John Wiley & Sons, Hoboken.
- [11] Srinivas, N. and Deb, K. (1994) Multiobjective Optimization Using Non-Dominated Sorting in Genetic Algorithms. *Evolutionary Computation*, **2**, 221-248. <http://dx.doi.org/10.1162/evco.1994.2.3.221>
- [12] Konstantinidis, A., Yang, K., Zhang, Q.F. and Zeinalipour-Yazti, D. (2010) A Multi-Objective Evolutionary Algorithm for the Deployment and Power Assignment Problem in Wireless Sensor Networks. *Computer Networks*, **54**, 960-976. <http://dx.doi.org/10.1016/j.comnet.2009.08.010>
- [13] Jia, J., Chen, J., Chang, G. and Tan, Z. (2009) Energy Efficient Coverage Control in Wireless Sensor Networks Based on Multi-Objective Genetic Algorithm. *Computers and Mathematics with Applications*, **57**, 1756-1766. <http://dx.doi.org/10.1016/j.camwa.2008.10.036>
- [14] Rajagopalan, R., Mohan, C.K., Varshney, P. and Mehrotra, K. (2005) Multi-Objective Mobile Agent Routing in Wireless Sensor Networks. *The 2005 IEEE Congress on Evolutionary Computation*, **2**, 1730-1737. <http://dx.doi.org/10.1109/cec.2005.1554897>
- [15] Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002) A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, **6**, 182-197. <http://dx.doi.org/10.1109/4235.996017>
- [16] Apaydin, T. and Ferhatosmanoglu, H. (2006) Access Structures for Angular Similarity Queries. *IEEE Transactions on Knowledge and Data Engineering*, **18**, 1512-1525. <http://dx.doi.org/10.1109/TKDE.2006.165>