Tai-hoon Kim Adrian Stoica
Ruay-Shiung Chang (Eds.)

Security-Enriched Urban Computing and Smart Grid

First International Conference, SUComS 2010
Daejeon, Korea, September 15-17, 2010
Proceedings

Springer

Volume Editors

Tai-hoon Kim
Hannam University
Daejeon, South Korea
E-mail: taihoonn@hannam.ac.kr

Adrian Stoica
Jet Propulsion Laboratory
Pasadena, CA, USA
E-mail: adrian.stoica@jpl.nasa.gov

Ruay-Shiung Chang
National Dong Hwa University
Shoufeng, Hualien, Taiwan, R.O.C.
E-mail: rschang@mail.ndhu.edu.tw

Library of Congress Control Number: 2010936201

CR Subject Classification (1998): C.2.4, J.3, C.2, H.4, I.4, I.5

ISSN 1865-0929
ISBN-10 3-642-16443-9 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-16443-9 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper 06/3180

Data Aggregation Using Mobile Agent Mechanism on Distributed Sensor Networks

Youn-Gyou Kook, Joon Lee, Ki-Seock Choi, Jae-Soo Kim, and R. Young-Chul Kim

Dept. of NTIS (National Science & Technology Information Service)
KISTI (Korea Institute of Science and Technology Information)
Daejeon, 305-806, Korea
{ykkook,rjlee98,choi,jaesoo}@kisti.re.kr
Dept. of Computer Information & Comm. Hongik University
Jochiwon, 339-701, Korea
bob@hongik.ac.kr

Abstract. In this paper, we propose mobile agent mechanism to efficiently aggregate data on distributed sensor networks (DSN), and describe the itinerary of mobile agent to establish by the sink node. The proposal of this mechanism is to reduce consumption of sensor network energy, network bandwidth and the limited resources. This mobile agent mechanism is based on mobile-agent-based DSN (MADSN) which the sink node migrate mobile agent to the sensor node on multicasting based on the binomial tree. It reduces time-cost to aggregate sensor data on MADSN.

1 Introduction

Since there are the limited resources, powers and network bandwidth on distributed sensor networks, it is important to aggregate sensor data in efficient on that environments. It needs to communicate with sensors for aggregating their data and transporting that to an application or some nodes. To efficiently aggregate sensor data in distributed sensor networks reduces consumption of sensor's power and network resources, and it can prolong life of sensor networks.

So many researchers proposed the methodology of prolonging sensor network's life, H. Qi suggested Mobile-Agent-based Distributed Sensor Networks (MADSN) as a solution of that[3, 5]. Mobile agent is small size programmed code which is able to migrate to the other nodes. That node has to be able to communicate with the other node and send/receive a mobile agent. MADSN is improved the traditional distributed sensor networks (DSN), that can migrate mobile agent in DSN. MADSN has the several benefits that are consumption of low power, extensibility, scalability and stability than DSN.

The sink node migrate mobile agent to the sensor nodes to aggregate sensor data in MADSN. There are the various methods of migration that the sink node migrate mobile agent to the all sensor nodes. The broadcasting is the sink node migrate mobile agent to the all sensor nodes one by one. The unicasting is same the ring topology that the sink node migrate mobile agent to the first sensor node with the itinerary to migrate the next node. And the multicasting is mixed migration method with

T.-h. Kim, A. Stoica, and R.-S. Chang (Eds.): SUComS 2010, CCIS 78, pp. 421–426, 2010.

broadcasting and unicasting[5,8,9]. The itinerary of multicasting is based on the binomial tree topology, the sink node migrate mobile agent some sensor nodes and a sensor node migrate that some sensor nodes. The binomial tree is the recursive tree structure, that is the optimization route path of mobile agent[1,8]. It reduces the time of migrating mobile agent and aggregating sensor data in efficient.

This paper proposes mobile agent mechanism based on the binomial tree. The itinerary of mobile agent is not static route path, and is to dynamically establish based on the location of sensor nodes and the priority of ones. To establish the dynamic itinerary of mobile agent, the sink node have to acquire the information of sensor node that are the level of tree, input node, input degree, sub tree, output degree and output nodes. This paper presents the method of finding the value of that to dynamically establish the itinerary of mobile agent and aggregating the sensor data in MADSN.

The organization of this paper is as follows. Section 2 descript MADSN and present the redesigned that. In section 3, we propose mobile agent mechanism based on the binomial tree. Section 4 presents the conclusion.

2 Mobile-Agent-based Distributed Sensor Network (MADSN)

In traditional DSNs, all of sensors collect sensor data and transmit data to the sink node or a higher-level processing element as like client-server paradigm. It decreases the life of sensor network to spend the power of sensors and to allocate the network bandwidth because large amounts of data are moved around the sensor network. And it is difficult to expand the scope of sensor networks that have to communicate sensors and sink node.

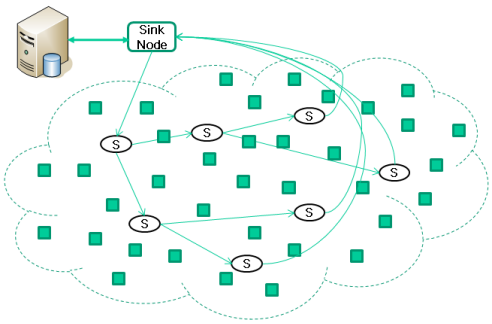

Fig. 1. The overview of mobile-agent-based distributed sensor network (MADSN)

So many researchers proposed the methodology of prolonging sensor networks, H. Qi designed MADSN and improved DSN architecture[5]. He presented the several important benefits as following that:

- Network bandwidth requirement is reduced. Mobile agent with small size moved around networks instead of round trip with large amount of raw data.
- Better network scalability. The architecture of MADSN is able to increase the number of sensor and expand the scope of network.

- Extensibility. Because mobile agent is small size code, we can improve the capability of sensor networks.
- Stability. If mobile agent migrates to the other node, then only the network connection is alive.

We redesigned the architecture of MADSN based on [5, 7], suggest mobile agent mechanism in those environments. The sink node communicates with the application server and launches mobile agent to the sensor node which is able to receive and send that. The sink node transmits the sensor data which is aggregated by mobile agent to the application server, and the application server sends a policy or the small size code (mobile agent) to aggregate sensor data.

3 Mobile Agent Mechanism

The itinerary of mobile agent in MADSN is based on the location of sensor and the priority of sensor nodes. It is important to order the route path of mobile agent which is comprised with sensor nodes topology based on the binomial tree to aggregate sensor data. Let n is the total number of sensor nodes in which have to visit for gathering sensor data. S is set of sensor nodes, as follows that: $S = \{S_{i-1} \mid i = 1,2,\cdots,n\},\ n < \infty$. S is the set of between S_0 to S_{n-1} that is composed of the sensor node's location. It is established dynamically the route path of sensor on which the sink node transports the mobile agent to aggregate sensor data. And P is set of the sensor node priorities, as follows that: $P = \{P_{i-1} \mid i = 1,2,\cdots,n\},\ n < \infty$. P is set of between P_0 to P_{n-1} that is composed of the sensor node priority which is established at the sink node to aggregate sensor data. Figure 2 shows the itinerary of mobile agent which is based on the binomial tree.

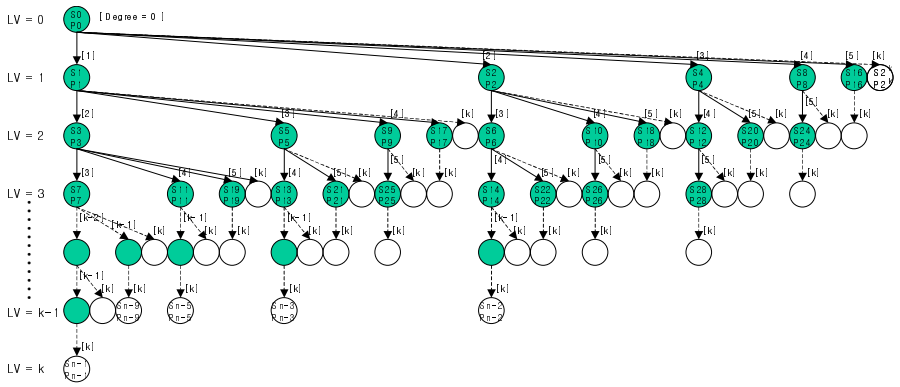

Fig. 2. The overview of binomial tree

So, S_i is composed of 6 elements which are LV_i, IN_i, ID_i, ST_i, OD_i, ON_i. The means of that is as follows: *LV* is a level of binomial tree, *IN* is an input node as the parent node, and *ID* is an input degree from the parent node. *ST* is a sub tree of node,

OD are an output degree to the child node, and *ON* is an output node to the child nodes. Where is a node S_i of the binomial tree *S*, $S_i \in S$, $S_i = \{LV_i, IN_i, ID_i, ST_i, OD_i, ON_i\}$. We defined the several variables to find the values of S_i's elements as follows that:

- n : the total number of sensor nodes to migrate the binomial tree
- l : the depth of a node at the binomial tree
- k : the maximum bit code size of a node
- m : the number of a node

If the total number of sensor nodes to migrate the binomial tree is *n* and the depth of that tree is *l*, then the formula of the binomial tree nodes is $2^l \leq (n-1) < 2^{l+1}$. We need to find the log value of that to define the maximum bit code size of sensor nodes.

$$2^l \leq (n-1) < 2^{l+1} \xrightarrow{\log} l \leq log_2(n-1) < l+1$$
$$\therefore k = l \tag{1}$$

And the order of mobile agent itinerary at the binomial tree is replaced by the number of a node *m*. The sink node establishes dynamically a set of *m* value, *m* value of S_i is not fixed. All of *m* value is progression 0 to *n-1* as below.

$$m = \prod_{i=0}^{k}(a_i * 2^i), \quad a_i = [0,1] \tag{2}$$

Through the values of k, m from (1), (2), the sink node acquires the elements of S_i. When m's value is 0, m is the first sensor node to visit by mobile agent. LV is same the depth of a node. R of IN means root node which is the first sensor node.

$$LV = l$$
$$IN = \begin{cases} R\ (m = 0) \\ m - 2^{k-1}\ (m \neq 0) \end{cases}$$
$$ID = \begin{cases} 0\ (m = 0) \\ l\ (m \neq 0) \end{cases}$$
$$ON = \begin{cases} 0\ (m = 0) \\ m + 2_{i=l}^{k}\ (m \neq 0 \end{cases}$$
$$ST = \begin{cases} k - l + 1\ (m = 0, ON \leq n) \\ k - l\ (m \neq 0, ON \leq n) \\ k - l - 1\ (m \neq 0, ON > n) \end{cases}$$
$$OD = \begin{cases} 0\ (k - l = 0) \\ log_2\left(m + 2_{i=0}^{k}\right)\ (k - l \neq 0, m = 0) \\ log_2(m + 2_{i=l}^{k})\ \ (k - l \neq 0, m \neq 0) \end{cases}$$

After the sink node acquires the elements of S_i, the sink node establishes the itinerary of mobile agent. The itinerary is composed of 'Sensor Node', 'Priority', 'Level', 'Input Node', 'Input Degree', 'Sub Tree', 'Output Degree', 'Output Node'. When the total number of sensor nodes is 54 and the sensor node number is same the priority, the sink node establishes the itinerary of mobile agent shown as figure 3.

```
54 Sensor Nodes : P0 is the first of all priority =>
SensorN PriorN  Level   InputN  InputD  SubTree OutputD OutputN
S0 [    P0,     0,      R,      0,      6,      6,      +1+2+4+8+16+32  ]
S1 [    P1,     1,      0,      1,      5,      6,      +3+5+9+17+33    ]
S2 [    P2,     1,      0,      2,      4,      6,      +6+10+18+34     ]
S3 [    P3,     2,      1,      2,      4,      6,      +7+11+19+35     ]
S4 [    P4,     1,      0,      3,      3,      6,      +12+20+36       ]
S5 [    P5,     2,      1,      3,      3,      6,      +13+21+37       ]
S6 [    P6,     2,      2,      3,      3,      6,      +14+22+38       ]
S7 [    P7,     3,      3,      3,      3,      6,      +15+23+39       ]
S8 [    P8,     1,      0,      4,      2,      6,      +24+40  ]
S9 [    P9,     2,      1,      4,      2,      6,      +25+41  ]
S10 [   P10,    2,      2,      4,      2,      6,      +26+42  ]
S11 [   P11,    3,      3,      4,      2,      6,      +27+43  ]
S12 [   P12,    2,      4,      4,      2,      6,      +28+44  ]
S13 [   P13,    3,      5,      4,      2,      6,      +29+45  ]
S14 [   P14,    3,      6,      4,      2,      6,      +30+46  ]
S15 [   P15,    4,      7,      4,      2,      6,      +31+47  ]
S16 [   P16,    1,      0,      5,      1,      6,      +48     ]
S17 [   P17,    2,      1,      5,      1,      6,      +49     ]
S18 [   P18,    2,      2,      5,      1,      6,      +50     ]
S19 [   P19,    3,      3,      5,      1,      6,      +51     ]
S20 [   P20,    2,      4,      5,      1,      6,      +52     ]
S21 [   P21,    3,      5,      5,      1,      6,      +53     ]
S22 [   P22,    3,      6,      5,      0,      0,      +T      ]
S23 [   P23,    4,      7,      5,      0,      0,      +T      ]
S24 [   P24,    2,      8,      5,      0,      0,      +T      ]
S25 [   P25,    3,      9,      5,      0,      0,      +T      ]
```

Fig. 3. An example of establishing the itinerary of mobile agent

We presented a mobile agent mechanism based on multicasting approach based on the binomial tree topology. The sink node send mobile agent to the sensor node of the itinerary. When the sink node migrate mobile agent, the itinerary of mobile agent is established dynamically by the sink node through the 6 elements of sensor node.

Let the total number of sensor node is 8 shown as figure 4. After the sink node establish the itinerary of mobile agent, the sink node transmit mobile agent to the first sensor node S_0. The sensor node S_0 migrate mobile agent to the other sensor nodes S_1, S_2, S_4 one by one. And sensor node S_1 migrate mobile agent to the child sensor node S_3, S_5 one by one. The leaf sensor nodes S_7, S_5, S_6, S_4 transmit mobile agent to the sink node shown as figure 4.

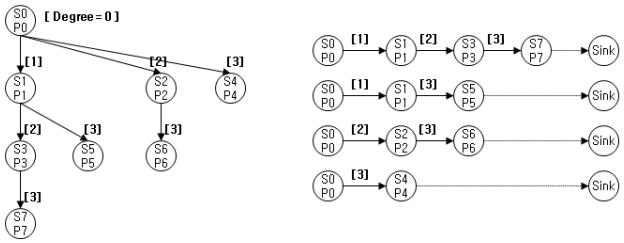

Fig. 4. Mobile agent mechanism based on the binomial tree

When a sensor node is fault, the itinerary of mobile agent is adjusted the priority of sensor nodes to guarantee the policy of fault-tolerance. A sensor node migrate mobile agent to the other node, after detecting the network channel. If a node S_j is fault, a node S_i adjusts the itinerary and migrate the next node S_k shown as figure 5.

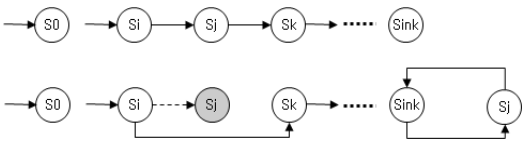

Fig. 5. Adjusting of sensor node priority

4 Conclusion

Since there are the limited resources, powers and network bandwidth on distributed sensor networks, it is important to aggregate data in efficient on that environments. It needs to communicate with sensors for aggregating their data and transport that to an application or some nodes. To effectively aggregate sensor data in distributed sensor networks reduces the power consumption of sensors and network resources, and it is able to bring sensor networks into all life-prolonging. Therefore, many researchers proposed the methodology of prolonging sensor networks, H. Qi suggested Mobile-Agent-based Distributed Sensor Networks (MADSN) as a solution of that.

So we suggested mobile agent mechanism to be reduced consumption of sensor network energy, network bandwidth and the limited resources. The proposed mobile agent mechanism is an approach of mobile agent migration which is multicasting based on the binomial tree. When the mobile agent launch to the MADSN, the sink node used the itinerary is composed of sensor nodes. All of the sensor nodes have the 6 elements to migrate mobile agent to the next sensor node. It is important to guarantee the stability of sensor networks because it provides the policy of fault-tolerance to migrate mobile agent.

References

[1] Kouvatsos, D., Mkwawa, I.-H., Awan, I.: One-to-All Broadcasting Scheme for Hypercubes with Background Traffic. In: Proceedings of the 36th Annual Simulation Symposium. IEEE CS, Los Alamitos (2003)

[2] Cappos, J., Hartman, J.H.: San fermın: aggregating large data sets using a binomial swap forest. In: USENIX Symposium on Networked System Design and Implementation (NSDI 2008), pp. 147–160 (2008)

[3] Qi, H., Wang, X., Sitharama Iyengar, S., Chakrabarty, K.: Multi-sensor Data Fusion in Distributed Sensor Networks Using Mobile Agents. Information Fusion (2001)

[4] Qi, H., Xu, Y., Wang, X.: Mobile-agent-based collaborative signal and information processing in sensor networks. Proceedings of IEEE, Special Issue on Sensor Networks and Applications 91(8), 1172–1183 (2003)

[5] Qi, H., Iyengar, S.S., Chakrabarty, K.: Multi-resolution Data Integration Using Mobile Agents in Distributed Sensor Networks. IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Rev. 31(3), 383–391 (2001)

[6] Roth, P.C., Arnold, D.C., Miller, B.P.: MRNet: A Software-Based Multicast/Reduction Network for Scalable Tools. In: Proceedings of the ACM/IEEE SC 2003 Conference (2003)

[7] Choi, S., Moon, S.J., Eom, Y.H., Kook, Y.-G., Jung, G.D., Choi, Y.G.: An Efficient Data Aggregation using Mobile Agent in Distributed Sensor Network. Proceedings of Korea Information Science Society 33(2)(B), 138–142 (2006)

[8] Cho, S.-H., Kim, Y.-H.: A Fast Transmission of Mobile Agents Using Binomial Trees. Korea Information Processing Society A 9-A(3) (September 2002)

[9] Kook, Y.-G., Young-Chul Kim, R., Choi, Y.-K.: Multi-Agent System Using G-XMDR for Data Synchronization in Pervasive Computing Environments. In: Lukose, D., Shi, Z. (eds.) PRIMA 2005. LNCS(LNAI), vol. 4078, pp. 297–309. Springer, Heidelberg (2009)