Balanced Itinerary Planning for Multiple Mobile Agents in Wireless Sensor Networks

Min Chen[1], Wei Cai[1], Sergio Gonzalez[2], and Victor C.M. Leung[2]

[1] Department of Computer Science and Engineering, Seoul National University, Korea
minchen@ieee.org, caiwei@mmlab.snu.ac.kr

[2] Department of Electrical and Computer Engineering, University of British Columbia, V6T 1Z4, Canada
{sergiog,vleung}@ece.ubc.ca

Abstract. In this paper, we consider the use of multiple mobile software agents to perform different tasks in wireless sensor networks (WSNs). To this regard, determining the number of mobile agents in the WSN remains an open issue in solving multi-agent itinerary planning (MIP) problem. We propose a novel scheme entitled MST-MIP based on minimum spanning tree, where each branch stemmed from the sink corresponds to a group of source nodes assigned for a mobile agent to visit. Furthermore, a balancing factor α is introduced to achieve a flexible trade-off control between energy cost and task duration, and the balancing MST-MIP algorithm is named BST-MIP. Extensive experiments show that MST-MIP has lower energy consumption than previous MIP proposals, while BST-MIP decreases the task duration up to 50%.

Keywords: mobile agent, itinerary planning, minimum spanning tree, wireless sensor networks.

1 Introduction

Compared to conventional data fusion in wireless sensor networks, the mobile agent (MA) system is better due to its intrinsic flexibility. In addition, it has been shown that data compression and fusion using MAs achieve better energy efficiency. However, using MAs also introduces larger task latency across a network with a large number of source nodes. To address this issue, a multi-agent system is proposed to achieve a balanced trade-off between energy cost and task latency.

In a multiple mobile software agents system, serval MAs roam in the network simultaneously. Each MA visits a subset of source nodes to retrieve information for the sink. In contrast to single MA itinerary planning (SIP), it is more challenging to determine the number of MAs and their corresponding subsets of source nodes for multi-agent itinerary planning (MIP), which is also called source grouping problem in this paper. To address this issue, we propose the use of minimum spanning tree (MST) to solve the MA grouping problem. We model the network topology as a totally connected graph (TCG). In order to simplify the TCG, only source nodes and the sink node constitute its vertices, while the weight of each arc can be basically estimated by the hop count among source nodes or the sink. According to such a hop-count-oriented TCG (H-TCG), we

J. Zheng, D. Simplot-Ryl, V.C.M. Leung (Eds.): ADHOCNETS 2010, LNICST 49, pp. 416–428, 2010.

can further generate a MST, where each branch stemmed from the sink corresponds to a group of sources included in the branch. Then, the number of branches in the MST is equivalent to the number of MAs used in the network, and each agent will collect sensory data by traversing the sources in its corresponding branch before returning to the sink. In this paper, we refer to the proposed solution for the MIP problem as MST-based MIP, which is denoted by MST-MIP. In MST-MIP, the critical issue is how to define the weight of each arc in the H-TCG. Intuitively, we can use the hop count of two vertices[1] in the H-TCG as the arc weight. Additionally, we further introduce a balancing factor α for the calculation of the weight. By adjusting α to suitable value, a balanced MST can be generated for the H-TCG. In this paper, the balanced MST-MIP algorithm is named BST-MIP, which can achieve a flexible trade-off control between energy cost and task duration. The importance of balanced source grouping in BST-MIP can be addressed in the following aspects:

- *Task duration*: Since multiple agents work in parallel, the task duration is mainly dependent on the delay incurred when an agent traverses along the branch containing the largest number of source nodes. Balancing the source grouping to eliminate the "bottleneck" branch will be the key to the reduction of task duration.
- *Lifetime*: From the perspective of the effective operation time before the first node depletes its energy, the nodes along a longer branch in the H-TCG will consume their energy more quickly than other nodes. Thus, a more balanced source grouping increases the lifetime by spreading the traffic load generated by multi-agent immigration more evenly in the whole network.

Extensive OPNET simulations are performed to show that the novel scheme outperforms the existing works. The reminder of the paper is organized as follows: Related work and a problem statement are introduced in Section 2 and Section 3. Then we describe the novel minimum spanning tree based source nodes grouping algorithm and its enhanced version in Section 4 and Section 5. The performance of our simulations will be analyzed in Section 6. Finally, Section 7 concludes this paper.

2 Related Work

Devising itinerary planning solutions for single MA is an essential part of MA research, which has been investigated by a number of researchers. Work in [2] proposes the simplest SIP solution: Local Closest First (LCF) and Global Closest First (GCF), whereas focus on energy efficiency is achieved by means of a genetic algorithm based solution presented in [3]. However, these approaches are not energy efficient as indicated in [4], in which the authors propose a better scheme named IEMF. In particular, IMEF denotes the importance of choosing the first visiting node. Based on this conclusion, it estimates energy costs of different choices of the first node and adopts the best solution to achieve energy efficiency.

However, the migration period of a single MA introduces larger end-to-end delay. Because of this, a multiple MAs system provides an alternative solution to achieve energy efficiency and reasonable task duration simultaneously. In CL-MIP [5], the authors

[1] Two vertices can be a pair of two sources or a pair of a source and the sink.

divide the MIP problem into two parts: source node grouping and a source node visiting sequence for each MA. They discover dense centers of source nodes and group them within a specific radius as one subset, each of which is assigned to a corresponding MA. Then, they determine the source node visiting sequence for each MA by employing IEMF. However, this paper still leaves an open research issue on choosing the optimal radius with the purpose of minimizing the total communication cost. Researchers in [6] also applied the genetic algorithm approach for the multiple MAs itinerary planning problem. Their work considers both the grouping and visiting sequence together to produce an energy-aware solution. The drawback of this is that the procedure of genetic evolution is complicated and hard to be realized in practice. In this paper, we focus on the trade-off between energy cost and task duration in MIP solutions and provide a novel grouping scheme base on minimum spanning tree theory.

3 Problem Statement

3.1 Grouping Problem

As shown in Fig.1, the data sink is denoted by the blue star. We also assume that the data sink is the information center with infinite power and sufficient computational capacity. Source nodes (denoted by green circles) are uniformly distributed in the deployment area. As seen in [2] [4] [5] [6], we assume geographical information of all source nodes stored in the sink, and that they remain static during MA migration. The dashed lines between the source nodes and data sink denote the distances, which will be utilized for the calculation of the estimated hop count.

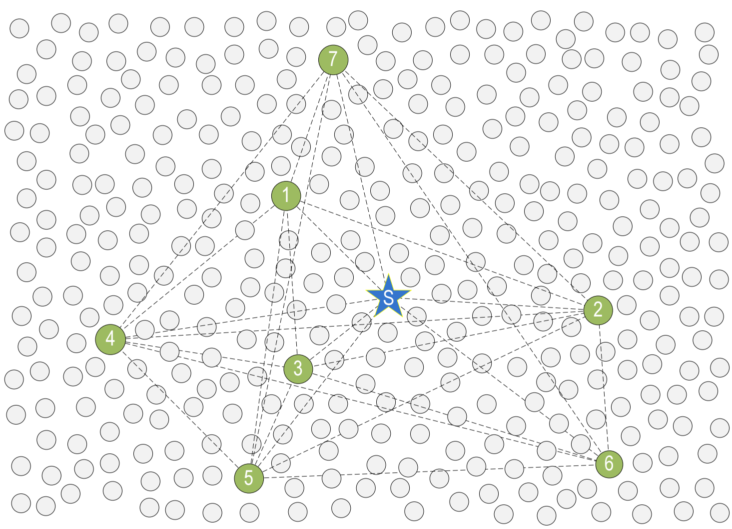

Fig. 1. A Typical Network Topology with Multiple Sources and Single Sink Node in WSNs

The key issue of solving the MIP problem is source-grouping, which includes two challenges: 1) determining the optimal number of MAs, and 2) assigning the corresponding subset of source nodes to each of them. The goal of source-grouping algorithm is to minimize task duration while decreasing total energy cost as possible.

3.2 The Metric to Evaluate the MA Migration Cost

In order to solve the grouping problem, we need to identify the key metric to evaluate the cost for MA's migration between two sources in terms of energy consumption and delay. In order to analyze the delay, let's consider the latency between two neighboring nodes first, which is the summation over the queuing, processing, propagation, and transmission delays:

– *Queuing delay*: since WSNs are normally assumed to support a low packet rate, communication traffic is considered to be rather low, thus queuing delay can be ignored.
– *Processing delay*: With respect to the processing delay, we assume that each node incurs similar delay to handle one MA.
– *Propagation delay*: This parameter can be neglected when compared to the other delays.
– *Transmission delay*: Because the size of an MA does not change between two source node, its transmission delay remains constant between any pair of intermediate sensor nodes.

Therefore, generally speaking, the delay taking place between any pair of intermediate nodes between two sources is similar. Consequently, the delay between two source nodes is proportional to the hop count between the two sources. On the other hand, the energy consumption for the migration of an MA between two sources is proportional to the number of transmissions, which is also proportional to the hop count. Thus, in order to estimate energy cost and/or delay, the metric to evaluate the weight between two sources can be simply the hop count. However, most of previous works [2, 3, 7, 8] ignore this issue, since they use distance between two sources as the weight between two sources.

3.3 Hop Count Estimation Formula

Compared to the distance, the estimated hop count can describe the energy cost more accurately, as discussed in Section 3.2. Thus, the issue of estimating hop count needs to be addressed at first. Assuming that there are two source nodes (i.e., i and j), let D_j^i denote the distance between the two sources, and let Let R represent the maximum transmission range for each hop. Since the actual hop distance between two nodes is smaller than R, we introduce a factor of $\xi, 0 < \xi \leq 1$ and let $R \times \xi$ represent the expected hop distance. Finally, let H_j^i denote the estimated hop count between i and j. Then, H_j^i can be estimated as follows:

$$H_j^i = \frac{D_j^i}{R \times \xi} \tag{1}$$

According to this equation, we can calculate estimated hop count between each pair of nodes.

4 Minimum Spanning Tree Based Source Grouping for MIP

In the section, we first introduce the construction of a minimum spanning tree (MST) based on the estimated hop count information, and propose an efficient source-grouping algorithm for multi-agent itinerary planning (MIP) in WSNs.

4.1 The Hop Count Based Minimum Spanning Tree

Total Connected Graph. We model the network topology as a totally connected graph (TCG). In order to simplify the TCG, only source nodes and the sink node constitute its vertices, while the weight of each arc can be basically estimated by the hop count among source nodes or the sink. Table.1 gives an example corresponding to Fig.1. Each element in the table represents the estimated hop count between two source nodes. The information from the table can be easily transformed into a hop count based total connected graph (H-TCG). In H-TCG, the vertices represent the source and sink nodes in the network, while the weight of each edge can be expressed at the corresponding estimated hop count.

Table 1. The hop count between nodes

	s	1	2	3	4	5	6	7
s	×	2	3	2	3	4	2	4
1		×	6	3	2	5	7	2
2			×	4	7	4	2	6
3				×	1	2	3	7
4					×	3	6	6
5						×	5	11
6							×	9
7								×

Definition of MST. Given a total connected graph $G = (V, E)$, we denote (u, v) as the edge connected to the vertex v and u; thus, $(u, v) \in E$. The weight of edge (u, v) is denoted as $w(u, v)$. If there is a subset T, which includes all of the vertices and has the feature that $w(T) = \sum_{(u,v)\in T} w(u, v)$ is minimum, then T is the minimum spanning tree of G.

Calculation of MST. There are a number of well-known algorithms that calculate the MST of a total connected graph. In our approach, we adopt one of the simplest approaches, known as the Prim Algorithm. Its pseudo-code is given as follows:

MST Based Node Grouping. In MST, there are several branches stemming from the sink. For each branch, a MA is dispatched to traverse the source nodes contained in the branch and return to the sink. Thus, in the proposed MST-MIP scheme, the number of MAs is equal to the number of direct vertices connected with the sink node. For each MA, the group of sources is determined by its corresponding branch.

As observed in the example shown in Fig.2, $(s, 1)$, $(s, 2)$ and $(s, 3)$ are three trunks originating at the sink node. $(s, 1)$ and $(1, 7)$ form the first branch; $(s, 2)$ and $(2, 6)$

Algorithm 1. Prim Algorithm for Minimum Spanning Tree

$T \Leftarrow \phi$
$V \Leftarrow \{Sink\}$
while $\exists(u \in V, v \notin V)$ **do**
  find (u, v) which has the minimum $w(u, v)$
  $T \Leftarrow T \cup (u, v)$
  $V \Leftarrow V \cup v$
end while
return T

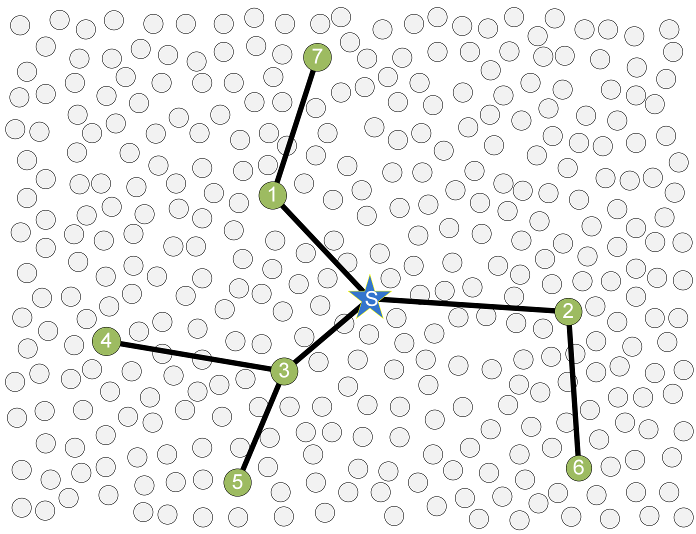

Fig. 2. An Illustrative Example of MST based Source-grouping

represent the second branch; $(s, 3)$ $(3, 4)$ and $(3, 5)$ constitute the third branch. Thus, three MAs are dispatched. One MA visits the source nodes 1, 7, the second MA visits the source nodes 2, 6, and the third MA will visit the source nodes 3, 4 and 5. After source-grouping, the visiting sequence in each subset of sources can be obtained by solving the SIP problem, which has been widely studied in previous work [2] [3] [4].

5 Balanced MST-MIP Algorithm

5.1 The Residual Problem

In the basic MST-MIP algorithm, the hop count of two vertices is directly set to the arc weight in the H-TCG, and the Prim algorithm with greedy feature is used to construct the MST. When the source nodes are close to each other, the relatively small hop count between two adjacent sources easily becomes the shortest edge during the selection of Prim algorithm. Consequently, the longer the branch is, the higher that the possibility

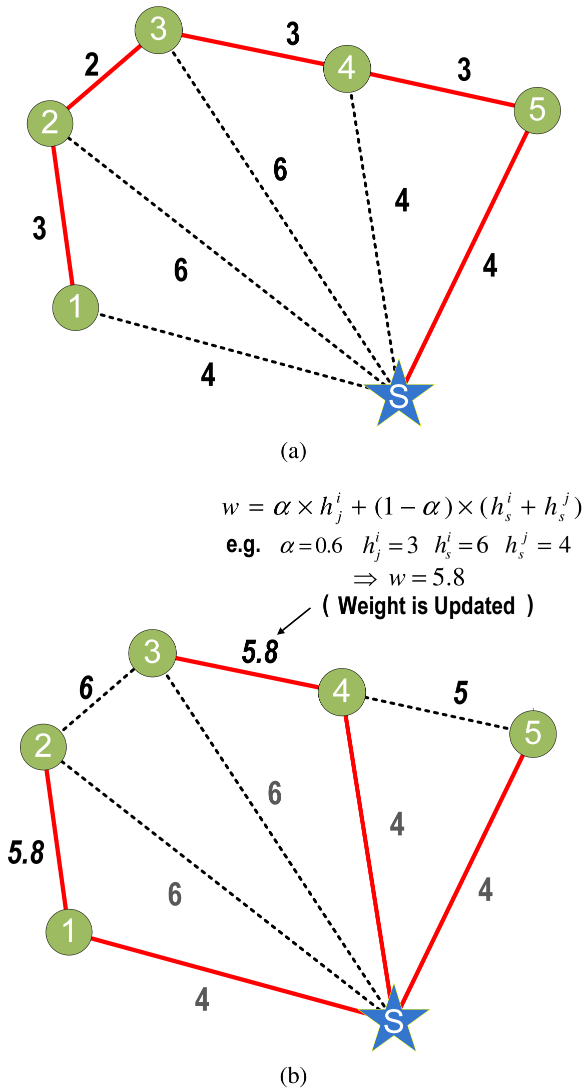

Fig. 3. MST-based MIP: (a) illustrative example of the problem in the basic MST-MIP; (b) an improved solution for source-grouping with $\alpha = 0.6$

to connect more source nodes is, which causes the reduction of the number of branches stemmed from the sink. Given the example shown in Fig.3(a), the MST contains a single branch, which means that only one MA is sent to the network, and that the task duration will be as high as in the SIP algorithm. Intuitively, the basic MST-MIP approach does not partition source nodes intelligently without considering the distribution of source nodes. Thus, we need to find a much better solution to achieve a balance between the energy cost and task duration.

5.2 Balancing Factor: α

Considering two source nodes i and j in the network, let us denote the estimated hop count between them as h_i^j, and denote their estimated hop count to the sink node as h_s^i and h_s^j, respectively. In order to address the unbalance issue existing in the basic MST-MIP algorithm, we introduce a balancing factor α to calculate the weights in the TCG as follows:

$$w = \alpha \times h_j^i + (1 - \alpha) \times (h_s^i + h_s^j) \tag{2}$$

where $\alpha \in [0, 1]$. By adjusting α to a suitable value, a balanced MST can be generated for the H-TCG. In this paper, the balanced MST-MIP algorithm is denoted by BST-MIP, which can achieve a flexible trade-off control between energy cost and task duration.

Given the example show in Fig. 3(b), if we set α to 0.6, the weight will be updated according to the relative distance between the source nodes and the sink, which produces a different minimum spanning tree. As the result, three MAs will be dispatched along three branches stemming from the sink, with the subsets of source nodes of $\{1, 2\}$, $\{3, 4\}$ and $\{5\}$. Compared to the MST generated in Fig. 3(a), the updated MST is more balanced.

6 Simulation and Analysis

6.1 Simulation Setup

We implemented the basic MST-MIP and BST-MIP schemes in the OPNET environment, and compare them to CL-MIP as presented in [5]. In the implementation, we define a WSN deployment area of $1000m \times 500m$ and allocate the sink node in the center of the network. Wireless sensors with a 802.11b/g network interface are uniformly distributed in the network. Random seeds are used to determine the position for the source nodes. For each MA, the parameters are set as shown in Table.2.

6.2 Evaluation Metrics

In order to evaluate the energy efficiency, task duration, and their overall performance from our simulation results, we consider the following three performance metrics, as reported previously [4] [5] [6]:

– Average Communication Energy: Used to indicate the total communication energy consumption in the network, including transmitting, receiving, retransmissions, overhearing and collision, to obtain each sensory data from all the target sources.
– Task Duration: Used for calculating the period for one particular task. For the case of the SIP algorithm, it is equivalent to the average end-to-end reported delay, which is the average delay from the time when a MA is dispatched by the sink to the time when it returns to the sink. For the case of the MIP algorithm, since multiple agents work in parallel, there must be one agent that returns to the sink at the end. Then, the task duration of the MIP algorithm is the delay of that agent.

Table 2. Simulation Parameters for MA

Raw Data Reduction Ratio	0.8
Aggregation Ratio	0.9
Raw Data Size	2048 bits
MA Code Size	1024 bits
MA Accessing Delay	10 ms
Data Processing Rate	50 Mbps

– Energy-Delay Product (EDP): Used for representing the overall performance from both the energy efficiency and task duration aspects. For time-sensitive applications over energy constrained WSNs, EDP (calculated by $EDP = energy \times delay$) gives us a unified view. The smaller this value is, the better the unified performance will be.

6.3 The Selection of a Balancing Factor α in BST-MIP

In this section, we study the impact of the factor α on energy cost, task duration and EDP, in order to discover its optimal value.

Fig.4 shows the impact of α on energy cost. When the factor α is smaller than 0.5, the network energy cost is stable at around 0.65 Joules/Task, which corresponds to the extreme case when all of the source nodes are connected to the sink directly in the MST, i.e., each source node is visited by an individual MA, and the benefits of the MA system in terms of data reduction and fusion are not utilized. Therefore, the energy cost is high. When the value of α is larger than 0.5, the impact of the distance between two sources on the arc weight increases, and a source node is easier to be included in an existing branch stemmed from the sink.

Fig.5 shows the impact of α on task duration. When the value of α is lower than 0.5, the impact of the distance between source and the sink on the arc weight increases, and a new branch stemming from the sink is more likely to be generated. Contrary to the

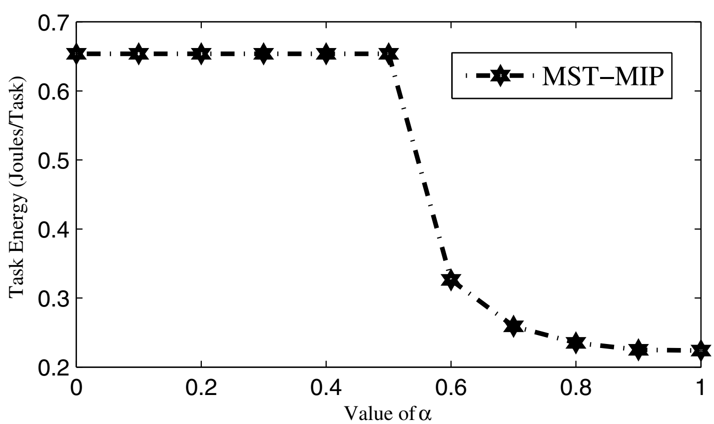

Fig. 4. The impact of α on energy cost

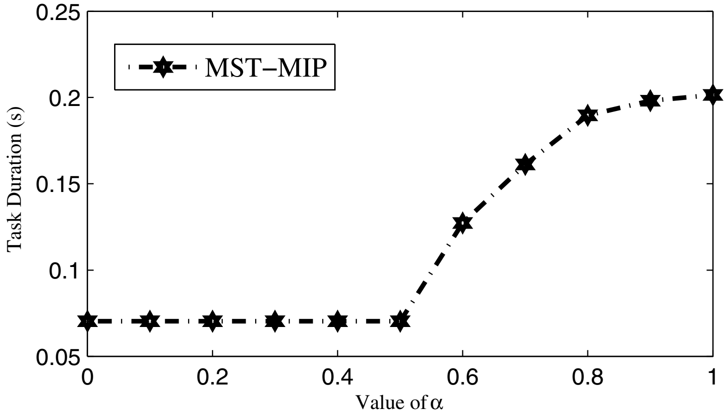

Fig. 5. The impact of α on task duration

energy cost, the task duration is relatively lower when the value of α is below 0.5. This is because the system delay is reduced for the task operation simultaneously performed by multiple MAs in parallel. When α increases from 0.5 to 1.0, the number of MAs decreases, which leads to a longer itinerary for each MA, and thus increasing the task duration as shown in Fig.5.

Similar to how Fig.4 and Fig.5 show the trade-off between energy cost and task duration, Fig.6 describes the impact of α on EDP, which is the metric that we employ to depict the overall performance. We observe that when α is below 0.5, EDP is stable according to the constant value of both energy cost and task duration in this interval.

When α increases, the value of EDP decreases until $\alpha = 0.6$, where EDP reaches its lowest value, indicating the best EDP performance. However, if we keep raising α from 0.6 to 1.0, the EDP raises back to a relative high value. From this study, we can conclude that 0.6 is the best value regarding to the EDP performance.

6.4 Performance Comparison

In this section, we compare the proposed MST-MIP and BST-MIP to two typical existing approaches, i.e., IEMF [4] and CL-MIP [5]. As a latest proposed solution for SIP, IEMF has the best performance in terms of energy cost and task duration, while CL-MIP is the first solution for the MIP problem. We changed the number of source nodes from 10 to 40 with the step of 5, and perform a series of simulations for each scheme. For a single data point, various random seeds are adopted, each of which corresponds to a scenario with different deployment of source nodes.

As shown in Fig.7, all of the three MIP schemes have lower task duration than IEMF, which verifies the effectiveness of using a multi-agent approach for reducing the system delay. The delay of CL-MIP and MST-MIP are comparable. However, BST-MIP achieves up to 50% reduction on the delay performance compared to CL-MIP and MST-MIP, which shows that BST-MIP allocates source nodes to multiple MAs in a more balanced fashion.

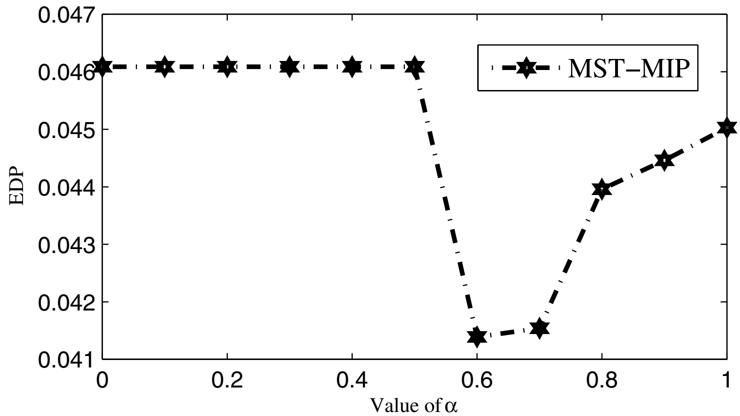

Fig. 6. The impact of α on EDP

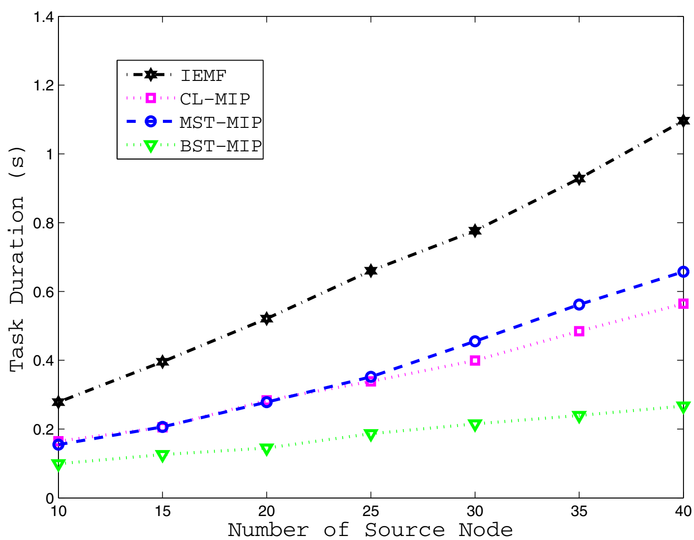

Fig. 7. The impact of the number of source nodes on task duration

However, when our focus is moved to the energy performance, it is observed that the overwhelming delay performance of BST-MIP is achieved by comprising some energy performance, and thus increasing about 15% energy cost compared to IEMF and CL-MIP. When compared to MST-MIP, BST-MIP requires 30% more energy cost if the number of source nodes is 40. This is because the fewer number of MAs used in MST-MIP, thus saving the communication overhead of delivering processing codes carried by a lager number of MAs.

Since MST-MIP and BST-MIP achieve the best performance in terms of energy cost and task duration, respectively, we need to further compare them through EDP

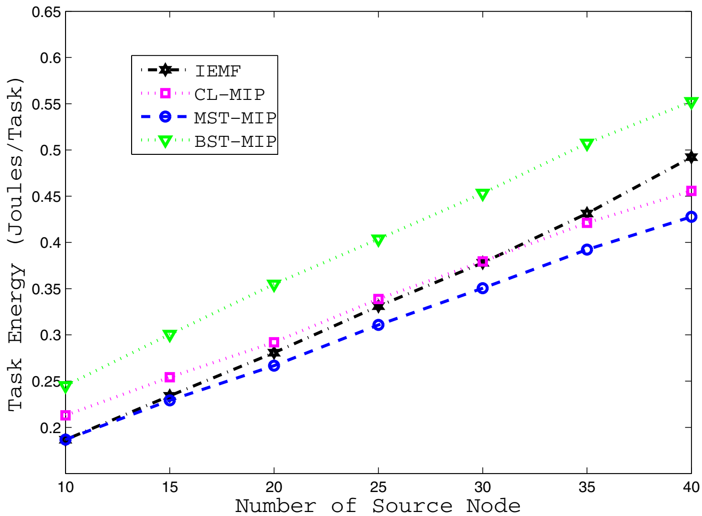

Fig. 8. The impact of the number of source nodes on energy cost

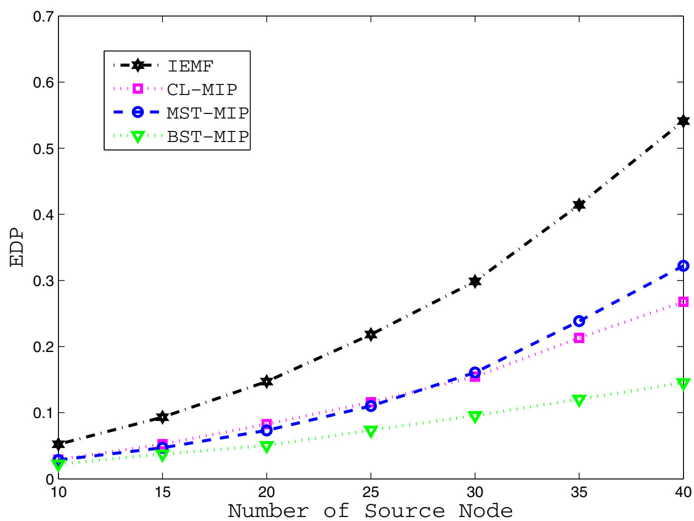

Fig. 9. The impact of the number of source nodes on EDP

performance. As shown in Fig. 9, BST-MIP has the least EDP, which illustrates that BST-MIP achieves efficient trade-off between energy and delay. When there are 40 source nodes, BST-MIP decreases EDP up to 70% compared to IEMF, and achieves a reduction of EDP up to 50% compared to CL-MIP and MST-MIP.

7 Conclusion

Compared to a single mobile agent system, the source-grouping problem is a key issue in planning itineraries for a multiple mobile agents system. In this paper, we first present a minimum spanning tree based source-grouping algorithm, and further propose the introduction of a balancing factor to achieve flexible trade-off control between energy cost and task duration. By adjusting the balancing factor, the QoS requirements in terms of delay can be satisfied for a large range of applications while reducing the energy cost to a maximum capacity. As part of our future work, we need to investigate a more efficient function to evaluate the arc weight for the construction of minimum spanning trees.

References

1. Chen, M., Gonzalez, S., Leung, V.: Applications and Design Issues of Mobile Agents in Wireless Sensor Networks. IEEE Wireless Communications Magazine (WCM), Special Issue on Wireless Sensor Networking 14(6), 20–26 (2007)
2. Qi, H., Wang, F.: Optimal itinerary analysis for mobile agents in ad hoc wireless sensor networks. In: Proceedings of the IEEE 2001 International Conference on Communications (ICC 2001), Helsinki, Finland (2001)
3. Wu, Q., Rao, N.S.V., Barhen, J., Iyengar, S.S., Vaishnavi, V.K., Qi, H., Chakrabarty, K.: On computing mobile agent routes for data fusion in distributed sensor networks. IEEE Trans. Knowledge and Data Engineering 16, 740–753 (2004)
4. Chen, M., Leung, V., Mao, S., Kwon, T., Li, M.: Energy-Efficient itinerary planning for mobile agents in wireless sensor networks. In: Proceedings of the IEEE 2009 International Conference on Communications (ICC 2009), Bresden, Germany, pp. 1–5 (2009)
5. Chen, M., Gonzlez, S., Zhang, Y., Leung, V.: Multi-agent itinerary planning for sensor networks. In: Proceedings of the IEEE 2009 International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine 2009), Las Palmas de Gran Canaria, Spain (2009)
6. Cai W., Chen M., Hara T., Shu L.: GA-MIP: Genetic Algorithm based Multiple Mobile Agents Itinerary Planning in Wireless Sensor Network. In: Proceeding of the 5th Annual International Wireless Internet Conference (WICON 2010), Singapore (March 2010)
7. Gavalas, D.: A Heuristic Algorithm for Designing Near-Optimal Mobile Agent Itineraries. Journal of Communications and Networks, IEEE/KICS 8(1), 123–131 (2006)
8. Gavalas, D., Mpitziopoulos, A., Pantziou, G., Konstantopoulos, C.: An Approach for Near-Optimal Distributed Data Fusion in Wireless Sensor Networks. In: Wireless Networks (2010), doi:10.1007/s11276-009-0211-0