Multi-Agent Itinerary Planning for Sensor Networks

1Min Chen, 2Sergio Gonzalez, 3 Yan Zhang, 2Victor C.M. Leung
1School of Comp Sci & Eng, Seoul National University, Korea (ubcmchen@gmail.com)
2Dept. of Elect & Comp Eng, Univ. of British Columbia, V6T 1Z4, Canada (sergiog.vleung@ece.ubc.ca)
3Simula Research Laboratory, Norway (yanzhang@ieee.org)

Agent-based data collection and aggregation have been proved to be efficient in wireless sensor networks (WSNs). While most of existing work focus on designing various single agent based itinerary planning (SIP) algorithms by considering energy-efficiency and/or aggregation efficiency, this paper identifies the drawbacks of this approach in large scale network, and proposes a solution through multi-agent based itinerary planning (MIP). A novel framework is presented to divide our MIP algorithm into four parts: visiting central location (VCL) selection algorithm, source-grouping algorithm, SIP algorithm and its iterative algorithm. Our simulation results have demonstrated that the proposed scheme lowers delay and improves the integrated energy-delay performance compared to the existing solutions with the similar computation complexity.

I. INTRODUCTION

The application-specific nature of a wireless sensor network (WSN) requires that sensor nodes have various capabilities. It would be impractical to store all the programs needed in the local memory of embedded sensors to run every possible application, due to the tight memory constraints. The intrinsically flexible features of mobile agent (MA) make it adaptable to diverse network conditions in dynamically reconfigurable WSNs.

An agent deployed in a sensor network is a special kind of software that migrates among network nodes to carry out a task autonomously, in order to achieve the objectives of the sink node.

Compared to its traditional client/server computing communications mechanism counterpart, mobile agent based computing has exhibited its unique efficiency in context-aware sensory environments [1], [2], [3], [4], [5], [6], [7], [8]. In a previous survey [1], we separated the agent design process for WSNs into four parts: architecture, itinerary planning, middleware system design, and agent cooperation for the design, development, and deployment of MA systems for high-level inference and surveillance in WSNs.

Among the four components, itinerary planning determines the order of nodes to be visited during agent migration, which has a significant impact on energy performance of the MA systems. Though the agent itinerary is critical to the network performance, it has been shown that finding an optimal itinerary is NP hard and still an open area of research. Therefore, heuristic algorithms [4], [2], [10] and genetic algorithms [5] are generally used to compute itineraries with a sub-optimal performance. Though our previously introduced IEMF and IEMA approaches [10] exhibit higher performance in terms of energy efficiency and delay compared to the existing solutions, the limitation of utilizing a single agent to perform the whole task, making the algorithm unsuitable in applications with a large number of source nodes needed to be visited. Typically, single agent itinerary planning algorithms have high efficiency in the applications with the following characteristics:

- The source nodes are distributed geographically close to each other.
- The number of source nodes is not large.

For a large scale sensor networks, with many nodes to be visited, single agent data dissemination exhibits the following pitfalls:

1) **Large Delay**: Extensive delay is needed when a single agent works for networks comprising hundreds of sensor nodes.

2) **Unbalanced load**: There are two kinds of unbalancing problems while using a single agent. First, in the perspective of the whole network, all of the traffic load is put on a single flow. Therefore, sensor nodes in the agent itinerary will deplete energy quickly than other nodes. Secondly, from the perspective of the itinerary, the agent size increases continuously while it visits source nodes, and so the agent transmissions will consume more energy in its itinerary back to the sink node.

3) **Insecurity with large accumulated size**: The increasing amount of data accumulated by the agent during its migration task increases its chances of being lost due to noise in the wireless medium. Thus, the longer the itinerary, the higher risky of the agent-based migration becomes.

In this paper, we propose a novel Multi-agent Itinerary Planning (MIP) algorithm to address the above issue. Traditionally, Single-agent Itinerary Planning (SIP) includes the following two challenges:

- Selecting the set of the source nodes to be visited by the mobile agent.
- Determining a node visiting sequence in an energy-efficient manner.

Compared to existing SIP proposals, the main contributions of this paper are listed as follows:
We introduce a novel source-grouping algorithm. Note in [11], clustering based architecture is utilized to facilitate mobile agent based data dissemination. Though our source-grouping algorithm partitions source nodes into several sets, which has a similar effect of grouping source nodes in clusters, we do not set up a hierarchical structure. Thus, our algorithm does not have any control message overhead for the clustering process.

- We propose an iterative algorithm for MIP solution.
- We propose a generic framework to design a MIP algorithm. Within this framework, any SIP algorithm can be extended to the corresponding MIP algorithm, where the SIP algorithm will be carried out iteratively until the source list is empty.

The remainder of the paper is organized as follows. The problem is stated in Section II. We present the proposed MIP algorithm in Section III. Our simulation studies are reported in Section IV. Section V concludes the paper.

II. PROBLEM STATEMENT

A. Motivation

In this section, the motivation for MIP proposal is illustrated through Eqs. (1) and (2). The agent size at the kth source depends on three parts: (1) the initial agent size ($l_{ma}^0$), which includes size of processing code and agent header; (2) size of reduced payload when visiting the first source node ($l_{data} \cdot (1 - r_1)$), where $r_1$ is the data reduction ratio at the first source. Note that there is no data aggregation at the first source; (3) accumulated size of the aggregated data payload after local processing from the second source node to the present source ($\sum_{i=2}^{k} l_{data} \cdot (1 - r_i) \cdot (1 - \rho_i)$). Thus, the final agent size increases linearly with the source number, as shown in Eqn. (1).

$$l_{ma}^k = l_{ma}^0 + l_{data} \cdot (1 - r_1) + \sum_{i=2}^{k} l_{data} \cdot (1 - r_i) \cdot (1 - \rho_i).$$  \hspace{1cm} (1)

$$E_{itinerary} = E_0^1 + \sum_{k=2}^{n} E_{k-1}^k(l_{ma}^{k-1}) + E_0^n.$$  \hspace{1cm} (2)

Fig. 1 presents a typical scenario of single agent based itinerary planning. In order to calculate the itinerary cost ($E_{itinerary}$), we divide the whole itinerary cost into three parts: (1) from the sink node to the first source node $S_1$, only the processing code and agent header are included in the MA packet. We denote the communication energy consumption in this part by $E_0^1$; (2) the second part starting from the time when MA leaves the first source node to the time when it visits the last source node $S_n$. The communication energy consumption in this phase is denoted by $\sum_{k=2}^{n} E_{k-1}^k(l_{ma}^{k-1})$, where $E_{k-1}^k(l_{ma}^{k-1})$ represent the communication energy cost for the MA to roam from source $k-1$ to source $k$ with agent size $l_{ma}^k$; (3) the third part starting from the time when MA finishes visiting all the source nodes to the time when it returns to the sink. The communication energy consumption in this part is denoted by $E_n^n$. Eqn.(2) shows that the itinerary cost is a squarely increasing function of the source node number, which causes the performance of the SIP algorithm to deteriorate in large scale sensor networks. The end-to-end agent delay exhibits a trend that is congruent to the similar trend as the itinerary cost. Thus, we are motivated to design a MIP algorithm that possesses the flexibility of adapting to the specific network parameters, such as network size, source node number, reduction ratio, aggregation ratio, sensor data size, etc. Specifically, a SIP algorithm can be deemed as a particular output of the SIP algorithm with a single agent.

B. A Generic Multi-agent Itinerary Planning Algorithm

We state our assumptions and define a generic MIP algorithm in this section as follows:

- primary itinerary design algorithms are executed at the sink, which has relatively plenty of resources in terms of energy and computation;
- the sink node knows the geographic information of all the source nodes. Note that in our algorithm, only source locations are needed, while the other algorithms [4], [5] need all of the nodes’ geographical positions.

In fact, the above assumptions are common in most of the solutions presented in [4], [5], [10] for the SIP problem. The

3MAs may deal with unexpected failures of arriving next source nodes, as soon as failure is detected, MA change the source destination node scheduled to be visited after the failed node.
previous SIP algorithms assume that the set of source nodes to visit is predetermined. In contrast, our MIP algorithm needs to group source nodes for different mobile agents, since the determination of source visiting set is a dynamic process. The proposed MIP algorithm can be deemed as the iterative version of a SIP solution, which can be divided into four parts:

- **Selection of Visiting Central Location (VCL) for an agent**: While using multiple agents, it is a challenging issue to use the least number of them while achieving the required coverage of source nodes. Strategically, the agent’s VCL is selected to the center of area with a high source node density. Finding an optimal agent number is also a NP-hard hard problem.

- **Determining the source visiting set**: In order to determining the source visiting set, we first isolate the visiting area, which is typically a circle/oval centered at the VCL and it has a certain radius. All of the source nodes in the disk will be included in the visiting list of the agent.

- **Determining a source-visited sequence**: This is the itinerary plan for the current agent. In this step, the problem is simplified into the Single-agent Itinerary Planning problem, whereby existing SIP solutions can be applied, such as LCF, GCF, MADD, IEMF and IEMA, etc.

- **Algorithm iteration**: If there are source uncovered source nodes, the next VCL will be calculated based on the remaining set of source nodes. The previous process will repeat until all of the source nodes have been assigned to a mobile agent.

### III. PROPOSED MIP ALGORITHM

1) **VCL-Selection Algorithm**: The basic idea of the proposed visiting central location (VCL) selection algorithm is to distribute each source’s impact factor to other source nodes. Let \( n \) denote the source number. Then, each source will receive \( n - 1 \) impact factors from other source nodes, and one from itself. After calculating the accumulated impact factor, the location of the source with the largest accumulated impact factor will be selected as VCL.

We achieve this by using the analogy of a gravity field: a source node is modeled by a small iron ball, and the network is seen as an elastic plane, as shown in Fig. 2(a). When the iron ball is put on the elastic plane, the plane will be naturally distorted to the shape as shown in Fig. 2(b). In our approach, we map the physical model to a sensor network in a way where each source will contribute with a certain gravity impact to a fixed location. If we overlap all of the source nodes’ gravity fields, there must be a location suffering the largest gravity. We define this location as VCL.

However, there are unlimited locations in the plane. Therefore, in order to reduce computational complexity, we make the following simplifications:

- the gravity impact is quantized by hop count between two source nodes.
- only the location of a source node is considered as a candidate to be selected as VCL.

We denote the set of \( n \) nodes by \( V_n \). For any two source nodes \( i, j \in V_n \), \( d_{ij} \) denotes the distance between \( i \) and \( j \). Then, we can estimate the hop count between \( i \) and \( j \) as \( H_i^j = \lceil d_{(i,j)} / R \rceil \), where \( R \) represents the maximum transmission range. To approximate the effect of a real gravity field, a gauss function is adopted to calculate the impact factor between \( i \) and \( j \):

\[
G_{ij} = e^{-\frac{(H_i^j-1)^2}{2\sigma^2}}. \tag{3}
\]

Fig. 3 shows an example with \( \sigma \) set to 8. A suitable setting should be heuristically selected for different network scale and different requirements of grouping effect.

The pseudo code of the VCL-selection algorithm is listed at Algorithm 1.

2) **Source-grouping algorithm**: Our source-grouping algorithm is very simple. Let \( A(VCL, R) \) denote the circular area centered at VCL with a radius of \( R \). Then, all of the source nodes within \( A(VCL, R) \) will be included in the visiting list which is assigned to the current agent. Algorithm 2 shows the pseudo-code of the proposed source-grouping algorithm.

3) **Iteration based MIP Algorithm**: For each iteration, a new VCL will be calculated for the remaining source list. Then, a new list of source nodes will be assigned to a mobile agent. To this moment, the itinerary for the agent can be planned by any SIP algorithms. In this paper, some typical SIP algorithms are tested, such as LCF, GCF and IEMF. If the remaining source list is not empty, the above process will repeat until all of the source nodes have been assigned to a mobile agent. The
The computational complexity of Algorithms 1, 2, and 3 is $O(n^2)$, and the one for our MIP scheme depends on the SIP algorithm. For example, if a SIP (e.g., LCF) has a computational complexity of $O(n^3)$, then a LCF-based MIP algorithm will have the same computational complexity.

IV. PERFORMANCE EVALUATION

A. Simulation Setting

We implement the proposed MIP algorithm as well as the three existing SIP algorithms (LCF, GCF and IEMF) using OPNET Modeler, and perform extensive simulations. We choose a network where nodes are uniformly deployed within a 1000m x 500m field. To verify the scaling property of our algorithms, we select a large-scale network with 800 nodes. We assume that the sink node is located at the right side of the field and multiple source nodes are randomly distributed in the network.

In all the figures presented in this section, each data point is the average of 25 simulation, which runs with different random seeds. The mobile agent specific parameters are shown in Table II.

B. Simulation Results

1) Multiple Itineraries Construction in the Proposed MIP Algorithm: In this section, we will show the snapshot of OPNET simulation for MIP algorithm, and a typical SIP

<table>
<thead>
<tr>
<th>TABLE II SIMULATION PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw Data Reduction Ratio ($\rho$)</td>
</tr>
<tr>
<td>Aggregation Ratio ($\rho$)</td>
</tr>
<tr>
<td>MA Accessing Delay ($\tau$)</td>
</tr>
<tr>
<td>Data Processing Rate ($V_p$)</td>
</tr>
<tr>
<td>Size of Sensing (Raw) Data ($l_{data}$) Default: 2048 bits</td>
</tr>
<tr>
<td>Size of Processing Code ($l_{proc}$)</td>
</tr>
<tr>
<td>The Number of Source Node (n)</td>
</tr>
<tr>
<td>Radius to the center point</td>
</tr>
</tbody>
</table>

The sensor application module consists of a constant-bit-rate source, which generates a sensor data report every 1 s (1024 bits each). As in [10], we use IEEE 802.11 DCF as the underlying MAC, and the radio transmission range is set to 60 m. The data rate of the wireless channel is 1 Mb/s. All messages are 64 bits in length. For consistency, we use the same energy consumption model as in [9]. The initial energy of each node is 5 Joules. The power consumptions for transmission, reception and idling are 0.66 W, 0.395 W, and 0.035 W, respectively. We count for all types of energy consumptions in the simulations, including transmission, reception, idling, overhearing, collisions and other unsuccessful transmissions, MAC layer headers, retransmissions, and RTS/CTS/ACKs.

We consider the following four performance metrics:

- **Task Duration**: in a SIP algorithm, it is the average delay from the time when a MA is dispatched by the sink to the time when the agent returns to the sink. In our MIP algorithm, since multiple agents work in parallel, there must be one agent which returns to the sink at last. Then, the task duration of our MIP algorithm is the delay of that agent.

- **Average Communication Energy**: the total communication energy consumption, including transmitting, receiving, retransmissions, overhearing and collision, over the total number of distinct reports received at the sink.

- **Hop Count**: in SIP, it is the average hop count of a mobile agent itinerary. In MIP, it is the accumulated hop counts of all the agents.

- **Integrated Performance**: For time-sensitive applications over energy constrained WSNs, we consider both delay and energy performances, and evaluate the integrated performance (denoted by $\eta$) in terms of task duration and average communication energy. The smaller the value of $\eta$ is, the better the integrated performance will be.

\[ \eta = \text{energy} \cdot \text{delay}. \]  
(4)
algorithm (i.e., LCF). Fig. 4 shows the result of source-grouping and itinerary planning for the first mobile agent. The circled node is selected as the first VCL, since it has the highest accumulated impact factor of 14.87. The second VCL has the highest impact factor of 4.14, which is much smaller than that of the first one. It is because the number of candidate source nodes is smaller than that of the first round selection, as shown in Fig. 5. Note that the number of left source nodes will become smaller and smaller, and only three source nodes are visited by the third agent, as shown in Fig. 6. Since the three agents are dispatched by the sink node in parallel, they collect sensory data concurrently. Observed from above figures, the third agent will return to the sink first, then the second agent. Finally, the task duration is actually the delay of the first agent. In order to compare MIP and SIP algorithms, the planned itinerary by LCF is plotted in Fig. 7. Intuitively, the length of itinerary is longer than that of the first agent in MIP algorithm. Thus, MIP algorithm will yield a shorter task duration than the SIP algorithm.

As shown in Fig. 8, MIP algorithm has absolute advantage in terms of task duration, which is only half of that of LCF. Note that the task duration of MIP is calculated fairly with SIP algorithms. Since we dispatch all of the mobile agents simultaneously, contention exists when multiple agents are close to each other. Even so, MIP still has superior delay performance than SIP algorithms.

In Fig. 9, the energy consumption of MIP algorithm is much higher than that of SIP algorithms when source number is small. Actually, it is only necessary for the usage of multi-
networks [12]. Fig. 11 shows that MIP algorithm has the best integrated performance, which verifies effectiveness of the proposed algorithm.

V. CONCLUSIONS

In this paper, we addressed the problem of itinerary planning for multi-agent based data dissemination, facilitating concurrent sensory data collection to reduce task duration extensively. The proposed multi-agent itinerary planning (MIP) algorithm has the similar complexity with most of single agent based itinerary (SIP) algorithm, and can be flexibly adaptive to network dynamics in various network scales. We will propose more efficient source-grouping algorithm in our future work.

Fig. 9. Task communication energy.

Fig. 10. Hop counts.

Fig. 11. Agent accumulated delay.

ACKNOWLEDGMENT

This work was supported in part by the Canadian Natural Sciences and Engineering Research Council under grant STPGP 322208-05. This work was supported in part by the IT R&D program of MKE/IITA [2007-F-038-03, Fundamental Technologies for the Future Internet] and grant number IITA-2009-C1090-0902-0006. OPNET University Program and The ICT at Seoul National University provides research facilities for this study.

REFERENCES