

Reducing Data Collection Latency in Wireless Sensor Networks with Mobile Elements

Liang He^{1,2}, Jianping Pan¹, and Jingdong Xu²

¹University of Victoria, Victoria, BC, Canada

²Nankai University, Tianjin, China

Abstract—The introduction of mobile elements has created a new dimension to reduce and balance energy consumption in wireless sensor networks, however, data collection latency may become higher. Thus the scheduling of mobile elements, i.e., how they traverse through the sensing field and when they collect data from which sensor, is of ultimate importance and has attracted increasing attention from the research community. Formulated as the Traveling Salesman Problem with Neighborhoods (TSPN) and due to its NP-hardness, so far only approximation and heuristic algorithms have appeared in the literature, but the former only have theoretical value now due to their large approximation factors. In this paper, following a progressive optimization approach, we propose a combine-skip-substitute (css) scheme, which is shown to outperform the best known heuristic algorithm. Besides the correctness and complexity analysis of the proposed scheme, we also show its performance and potentials for further extension through extensive simulation results.

Keywords—Wireless sensor networks, mobile elements, TSPN

I. INTRODUCTION

Collecting data from the nodes deployed in the sensing field is one of the main applications of wireless sensor networks [1]. Typically, data collection only relies on wireless communications between sensor nodes and the sink node, which may suffer from the following problems. First, wireless communications, especially long-range ones, may consume the limited onboard energy supply of sensor nodes excessively. Second, even if shorter-range, multi-hop wireless communications are adopted, due to the data aggregation towards the sink, nodes around it still have to consume energy much faster than others, leading to a lower overall network lifetime. Mitigation has appeared in the literature, such as nonuniform topology [2], transmission power [3], media access [4] and routing control mechanisms [5], but the intrinsic high and unbalanced energy consumption still remains as a main challenge.

Another approach to data collection in wireless sensor networks utilizes the often-available, controlled mobility of certain nodes, referred to as *mobile elements* in this paper. For example, in an undersea observatory such as the UVic NEPTUNE [6], underwater autonomous vehicles can cruise through several experimentation sites, talking to experiment devices through very-short-range, high-data-rate optical communication technologies, and bring the data back to the junction boxes, which are forwarded to the shore station through the cabled network. Even though experiment devices may have relatively longer-range acoustic communication capabilities, the achievable data rate in such a harsh environment is very

low. Underwater robots are also used to track the latest Mexican Gulf oil leak and predict where it has headed [7]. Similar scenarios have appeared in structural health monitoring, where a radio-controlled helicopters collecting data from large-scale civil infrastructures [8]. By utilizing mobile elements, not only more energy can be conserved and balanced on sensor nodes, but also the communications and networking becomes possible in very sparse networks with “store-carry-forward.”

Although attractive, data collection with mobile elements in wireless sensor networks still poses its own challenges. Due to the relatively lower speed of mobile elements when compared with electromagnetic or acoustic waves, data collection may suffer a higher latency than multi-hop forwarding when the latter is feasible at high energy cost [9]. The latency, mainly determined by the mobility and scheduling of mobile elements, i.e., how they traverse through the sensing field and when they collect data from which sensor, is the main focus of the research efforts on this topic and that of this paper. In this paper, we tackle this problem from a new angle. We follow a progressive optimization approach, trying to reduce the tour length, thus the travel time, of mobile elements gradually through *combining* the collection sites for nearby sensor nodes (or cluster heads) and then *skipping* and *substituting* (i.e., *css*) certain sites. We prove the correctness of the progressive optimization approach, and show the performance of this approach through extensive simulation results.

The contributions of this paper are threefold. Through the formulation of such problems as the Traveling Salesman Problem with Neighborhoods (TSPN) and due to its NP-hardness, several approximation and heuristic algorithms have appeared in the literature, but so far the best approximation algorithms only have theoretical value due to their large approximation factors, and we have shown that our progressive optimization-based *css* scheme can outperform the best known heuristic algorithm in the literature. Second, the correctness of the scheme has been proved and its complexity analysis been provided. Third, we have shown its effectiveness and efficiency through extensive simulation results.

The rest of this paper is organized as follows. The existing work related to exploring mobility for data collection in wireless sensor networks is reviewed in Section II. In Section III, we define the scope of our problem and highlight our approach. In Section IV, we present the *css* scheme, which progressively reduces the tour length through *combining*, *skipping* and *substituting*. Its performance is evaluated in

Section V. Further discussion is offered in Section VI. Finally, we conclude this paper in Section VII.

II. RELATED WORK

Recently, many research efforts have appeared in the literature to explore the mobility in wireless sensor networks for data collection, we only survey the most related ones here [9]–[14]. The mobility-assisted data collection was classified into three categories in [15]: with random mobility [10], predictable mobility [11], and controlled mobility [14], respectively. Our work falls into the last category.

[10] is a pioneer work on this topic, where a three-tier network architecture was proposed. The mobile entities, referred to as Data Mobile Ubiquitous LAN Extensions (MULEs), lie in the middle tier on top of the stationary sensor nodes, move around in the network to collect data from sensor nodes, and ultimately upload the data to the sink. The term *Data MULEs* was widely used in the literature since then. In [11], the data collection process with predictable mobility was modeled as a queuing system, and the success of data collection was analyzed based on it. In [14], a mobile data observer, called SenCar, was used as a mobile base-station in the network. It also showed that the design of the traveling tour is critical for SenCar to accomplish data collection jobs successfully.

Observing the importance of the traveling tour, a lot of efforts were put into its optimal design, e.g., [9], [12], [13]. The tour selection problem can be modeled as a *Traveling Salesman Problem with Neighborhoods* (TSPN), an NP-hard problem, if we do not consider the data rate constraints between the mobile element (ME) and sensor nodes, where all the *neighborhoods* are possibly intersected communication disks. It has been proven that approximating Euclidean TSPN within a factor of $(2-\varepsilon)$ is also NP-hard [16]. The performance of the existing approximation algorithms for TSPN has only been characterized theoretically in terms of approximation factors, which are often quite large. Specifically, for the case of possibly intersected equal-sized disks, the best result so far was given in [17], where an approximation factor of 11.15 is achieved. In our scenarios, knowing such a loose bound is obviously of little practical value. Both [12] and [13] took this TSPN approach to obtain the tour. In [12], the authors started with an optimal TSP tour, based on which they reduced the problem's search space, and adopted three evolutionary algorithms to obtain the traveling tour. The case where multiple MEs exist in the network was considered in [13]. On the other hand, the problem of tour selection for mobile elements was formulated as the *Label-Covering Problem* in [9], which was also proved to be NP-hard. An heuristic algorithm using dynamic programming was presented there to solve the problem, which was shown through simulation to be able to achieve better results than [14], [18], [19] and is considered as the best known heuristic algorithm.

Utilizing the fact that the communication disks of sensor nodes may intersect with each other, we propose the css scheme in this paper, which greatly shortens the tour by

combining several data collection jobs together when possible and further skipping and substituting some collection sites.

III. PRELIMINARIES

In this section, we first list the notations used in this paper, and then give the scope of our problem definition and highlight our approach. The scheme is detailed in the next section.

- L : the side length of the square sensing field;
- $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$: the set of n sensor nodes with corresponding location $\{l_i\}$ where $i = 1, 2, \dots, n$;
- B : the base station of the network, with location l_0 ;
- v : the constant speed of the ME;
- d : the communication range between the ME and sensor nodes;
- \mathcal{T} : the set of all possible tours that start and end at l_0 ;
- T_{tsp} : the optimal tour with length $|T_{tsp}|$ of the TSP problem, which connects l_i and $i = 0, 1, 2, \dots, n$;
- T_{com} : the traveling tour with length $|T_{com}|$ obtained after the combination algorithm, which connects location l_0 and collection sites l'_i and $i = 1, 2, \dots, n'$;
- T_{css} : the traveling tour with length $|T_{css}|$ obtained after the entire css scheme, which connects locations l_0 and collection sites l''_i and $i = 1, 2, \dots, n''$;
- T^* : the optimal traveling tour with length $|T^*|$;
- \overline{xy} : the sub-tour that connects collection sites x and y directly;
- δ : the control parameter used in the substitute algorithm.

Our goal is to find the shortest tour to collect data from all n sensor nodes (or cluster heads) in the shortest time. We mainly consider one single-radio ME moving between collection sites directly without obstacles. The simplified problem is still NP-hard [9] and we offer some discussion on how to extend the scheme to other scenarios in Section VI. Our approach is to achieve the goal progressively by starting from a TSP problem formulation and considering the advantage of wireless communications through the css scheme. Focusing on the offline scenario, we assume the ME has the location information of sensor nodes or cluster heads, computes the traveling tours at the stage of network planning, and follows the tour to collect data. Even in the cases where location information is not available or changes dynamically over the time, our results still can be a performance bound for data collection latency. Furthermore, the css scheme can be extended to apply to the online scenario, where data collection requests arrives progressively, based on the requests currently available.

IV. TRAVELING TOUR W/O DATA RATE CONSTRAINTS

In this section, we consider the case with a fixed communication range between the ME and sensor nodes without data rate constraints to introduce the css scheme, and then evaluate and compare its performance through simulation, besides the analysis on its correctness and complexity.

A. Problem Formulation

We assume the *unit disk* communication model, and the time required for data transfer between the ME and sensor

nodes is negligible when compared with the traveling time of the ME [20]. With this assumption, all the data collection jobs can be accomplished as long as the traveling tour intersects with the communication disks of all sensor nodes. We call a traveling tour *feasible* if all data collection jobs can be accomplished when the ME travels along it. The tour selection problem in this case can be formulated as:

$$\min_{T \in \mathcal{T}} |T| \quad \text{s.t.} \quad \forall s_i \in \mathcal{S}, \exists e \in T, |s_i, e| \leq d, \quad (1)$$

where $|T|$ is the tour length and $|s_i, e|$ is the shortest Euclidean distance from s_i to any path segment e in T , i.e., all sensor nodes are *path-covered* by T within d . Conversely, denote $\mathcal{C}(e)$ as the set of the sensor nodes that are path-covered by e .

B. Combine-Skip-Substitute Scheme

Utilizing the nonzero wireless communication range between the ME and sensor nodes and the fact that the communication disks of nearby sensor nodes may intersect with each other, the css scheme employs three steps to progressively shorten the ME's traveling tour: it starts with an optimal TSP tour based on the set of sensor nodes in the sensing field, then it combines the data collection sites by a modified Welzl's algorithm when possible, and finally it uses the skip-and-substitute algorithm to further shorten the tour.

1) *Find the Optimal TSP Tour*: We adopt an existing TSP solver, Concorde, to obtain the optimal TSP tour, i.e., T_{tsp} [21]. The Concorde TSP Solver uses an exact algorithm for TSP, which follows the cutting-plane method, iteratively solving the LP relaxations of the TSP problem. It has been used to obtain the optimal solutions for 107 of the 110 TSPLIB instances, among which the largest one has 15,112 cities. The efficiency of Concorde has been testified by many experiments, e.g., Concorde can solve a TSP problem with 120 cities in 3.3 seconds with a 400 MHz CPU [22].

In our problem setting, the order of serving data collection jobs is determined by T_{tsp} , which reduces the search space greatly [12]. By definition, T_{tsp} is always feasible. We then conduct the combine, skip, and substitute operations based on this order, with communication range d .

2) *Combine Collection Sites by Modified Welzl's Algorithm*: Several data collection jobs can be combined if the corresponding sensor nodes are close to each other geographically, and the ME can carry out these jobs at a single collection site. Enlightened by this intuition, the css scheme first reduces the number of collection sites that the ME has to visit along the T_{tsp} obtained above, by adopting a modified version of *Welzl's algorithm* [23], to combine the collection jobs of nearby sensor nodes on T_{tsp} into a new collection site when possible.

The Welzl's algorithm computes the smallest enclosing disk of a finite set of points on the plane in a linear expected time, and returns the radius and center of the disk. We adopt the Welzl's algorithm in a different way, which we refer to as the *modified Welzl's algorithm*, to combine sensor locations within a radius of d whenever possible. The modified Welzl's algorithm is shown in Algorithm 1, which returns the smallest

enclosing disk of a given *subset* of sensor nodes if its radius is no more than d , or *false* otherwise.

Algorithm 1 Modified Welzl's Algorithm

Input: a subset of sensor nodes \mathcal{S}' , and the communication range d between the ME and sensor nodes;
Output: if the subset can be covered by a disk with radius at most d , return the disk's center and radius, or *false* otherwise.
 $radius \leftarrow \infty$; $center \leftarrow \emptyset$;
 $(radius, center) = \text{Welzl}(\mathcal{S}')$; //Welzl's algorithm on \mathcal{S}'
if $radius > d$ **then**
 return *false*;
else
 return $radius$ and $center$.
end if

With the modified Welzl's algorithm, we can carry out the combination operation as in Algorithm 2. Essentially, we check whether the nearby sensor nodes on T_{tsp} can be covered by a disk of radius at most d . If so, instead of visiting all covered sensor nodes, the ME can simply visit the center of the enclosing disk as the collection site, and collect data from all covered sensor nodes through wireless communications.

Algorithm 2 Combination by modified Welzl's Algorithm

Input: the set of sensor nodes \mathcal{S} , and the communication range d between the ME and sensor nodes;
Output: a traveling tour $T_{com} \in \mathcal{T}$ that intersects with the communication disks of all sensor nodes.

obtain the optimal TSP tour by Concorde for \mathcal{S} : i.e., $T_{tsp} = \langle l_0, l_1, \dots, l_n, l_0 \rangle$;
while there exist intersected disks **do**
 for all l_i ($i = 1, 2, \dots, n-1$) **do**
 find the maximum j ($i \leq j \leq n$) by the modified Welzl's algorithm, such that all the locations in $\{l_i, l_{i+1}, \dots, l_j\}$ are not involved in previous combination operations, and can be covered by a disk with radius no more than d ;
 $N_i \leftarrow j - i + 1$;
 end for
 select l_i with maximum N_i ;
 combine $\{l_i, l_{i+1}, \dots, l_{i+N_i-1}\}$ into c in T_{tsp} , where c is the center of the smallest enclosing disk that covers $\{l_i, l_{i+1}, \dots, l_{i+N_i-1}\}$;
end while
 $T_{com} \leftarrow T_{tsp}$;
return T_{com} .

Theorem 1. *Without data rate constraints, T_{com} is feasible.*

Proof: For the sensor nodes whose collection jobs have not been combined, they are still on T_{com} . For the sensor nodes whose collection jobs have been combined together, there exists a new collection site on T_{com} , from which is at most d to these sensor nodes. Therefore, T_{com} still intersects with the communication disks of all sensor nodes as T_{tsp} . ■

Theorem 2. $|T_{tsp}| \geq |T_{com}|$.

The equality holds if and only if none of the locations on T_{tsp} can be combined, e.g., in an extremely sparse network. Proof is omitted due to space constraints.

Definition. If collection site l' in T_{com} is obtained by combining $\{l_i, l_{i+1}, \dots, l_{i+k-1}\}$ in T_{tsp} , we say that l' represents these locations in T_{com} , denoted as $\mathcal{R}(l') = \{l_i, l_{i+1}, \dots, l_{i+k-1}\}$. $\mathcal{R}(l') = l'$ if l' is not obtained by combination.

3) *Skip and Substitute Through Binary Search:* The basic idea of skip-and-substitute is that, for each collection sites l'_i in T_{com} , we first try to skip l'_i by going directly from l'_{i-1} to l'_{i+1} , if all locations in $\mathcal{R}(l'_i)$ are path-covered by $\overline{l'_{i-1}l'_{i+1}}$. Otherwise, we select another collection site l''_i on $\overline{l'_i l'_{i+1}}$, which is done by binary search, and all locations in $\mathcal{R}(l'_i)$ are covered by $\overline{l'_{i-1}l''_i}$. A control parameter δ is used to determine when to terminate the binary search.

Algorithm 3 Skip-and-Substitute through Binary Search

Input: $T_{com} = \langle l_0, l'_1, l'_2, \dots, l'_{n'}, l_0 \rangle$, the communication range d between the ME and sensor nodes, and the control parameter δ for binary search;

Output: a further shortened traveling tour T_{css} .

repeat

 for all l'_i ($i = 1, 2, \dots, n'$) do

 if all collection sites that are in both $\mathcal{R}(l'_i)$ and $\mathcal{C}(\overline{l'_{i-1}l'_i})$ are also in $\mathcal{C}(\overline{l'_{i-1}l'_{i+1}})$ and all collection sites that are in $\mathcal{R}(l'_{i+1})$ are also in $\mathcal{C}(\overline{l'_{i-1}l'_{i+1}})$ then

 skip l'_i from T_{com} ;

 for all $\forall p \in \mathcal{R}(l'_i)$ do

$\mathcal{R}(l'_{i+1}) \leftarrow p$;

 end for

 else

$start \leftarrow l'_i$; $end \leftarrow l'_{i+1}$;

 while $|start, end| > \delta$ do

$q \leftarrow \text{midpoint}(start, end)$;

 if all collection sites that are in both $\mathcal{R}(l'_i)$ and $\mathcal{C}(\overline{l'_{i-1}l'_i}) \cup \mathcal{C}(\overline{l'_i q})$ are also in $\mathcal{C}(\overline{l'_{i-1}q})$ and all collection sites that are in both $\mathcal{R}(l'_{i+1})$ and $\mathcal{C}(\overline{l'_{i-1}l'_i}) \cup \mathcal{C}(\overline{l'_i q})$ are also in $\mathcal{C}(\overline{l'_{i-1}q})$ then

$start \leftarrow q$;

 else

$end \leftarrow q$;

 end if

 end while

 substitute l'_i by q in T_{com} ;

 end if

 end for

until T_{com} cannot be further shortened

$T_{css} \leftarrow T_{com}$;

return T_{css} .

Following the same argument as Theorem 1, we have

Theorem 3. Without data rate constraints, T_{css} is feasible.

Since we replace two adjacent paths by one direct path, by *triangle inequality*, we have the following two lemmas. Due to space constraints, their proofs are omitted here.

Lemma 1. Each skip operation reduces $|T_{com}|$, or leaves $|T_{com}|$ unchanged.

Lemma 2. Each substitution operation reduces $|T_{com}|$.

Directly following these two lemmas, we have

Theorem 4. $|T_{com}| \geq |T_{css}|$.

Figure 1 illustrates the general idea of the css scheme. For example, a TSP tour $\langle B, 1, 2, 3, 4, 5, 6, 7, B \rangle$ is first established. With the combination algorithms, 3 and 4 are combined into A , and 5, 6 and 7 are combined into C . Following the shortened tour $\langle B, 1, 2, A, C, B \rangle$, we can skip 1 since it is path-covered by $\overline{B2}$, and we can substitute A by A' , since $\overline{2A'}$ path-covers 3 and 4 as well. Similarly, C is substituted by C' to path-cover 5, 6 and 7. The progressive optimization will substitute 2 by D to further shorten the tour.

Although we use the uniform communication ranges for all sensor nodes to describe the css scheme, the scheme also applies to the case where the communication ranges are different, since we can easily consider the different but fixed communication ranges when determining whether a sensor node is path-covered by any segments of the tour.

C. Performance Analysis

1) *Upper-bound of the Tour Length:* In [17], it has been shown that for a TSPN problem of n points with disjoint unit disks of radius d , we have $|T_{tsp}| \leq |T^*| + 2nd$. This bound is obtained in the case where no combination is made. Following the same idea, we can extend it to the case with possible combination operations, which yields an even tighter bound due to a smaller number of collection sites after combination.

Theorem 5. $|T_{css}| \leq |T_{com}| \leq |T^*| + 2n'd$.

2) *Lower-bound of the Tour Length:* Suppose the optimal TSP tour is known in a unit square with n points. Given a random point on the tour, we can divide the tour into two parts: the forward sub-path and the backward sub-path, both of which contains approximately $n/2$ points, and we know a lower bound of the TSP is $\sqrt{n/2}$ [24]. If we treat the $n'' < n$ collection sites after the css scheme as a TSP problem,

$$|T_{css}| \geq L\sqrt{n''/2}. \quad (2)$$

3) *Time Complexity of the Scheme:* Denote C_{tsp} as the time complexity to obtain the optimal TSP tour. The time complexity of the combination algorithm is $C_{tsp} + O(n^3 \log n)$. For the skip-and-substitute algorithm, the time complexity is $O(n^2 \log \frac{1}{\delta})$. Thus the time complexity of the entire css scheme is $C_{tsp} + O(n^3 \log n) + O(n^2 \log \frac{1}{\delta})$. We will show the time for skip-and-substitute is relatively small when compared with that for combination in the sequel.

V. PERFORMANCE EVALUATION

We evaluate the performance of the css scheme and compare it with the Label-Covering algorithm [9] in this section. Based on the parameters from the real systems in [25], and observing the fact that the mobility-assisted data collection is especially

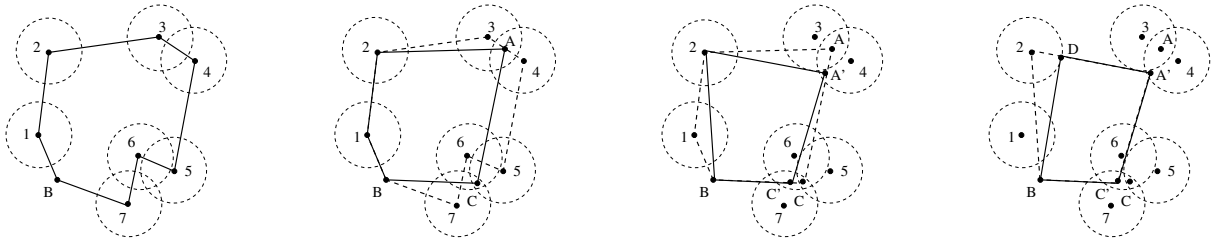


Fig. 1. Demonstration of the css scheme.

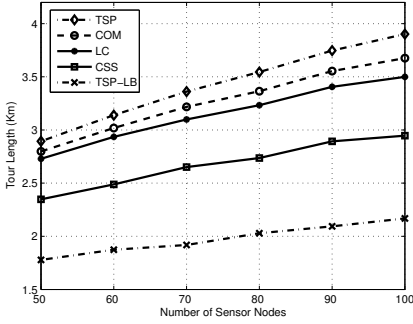


Fig. 2. Tour length with different numbers of sensor nodes.

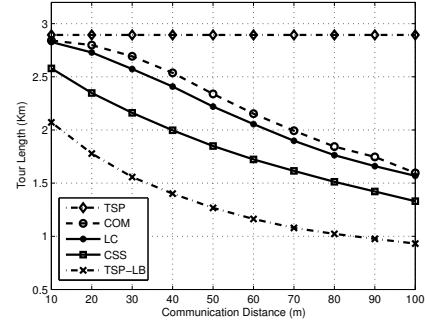


Fig. 3. Tour length with different communication distances.

important in sparse and not always connected networks, we consider a sparse square sensing field with size $500 \times 500 m^2$, where nodes are uniformly deployed at random, and the constant ME speed is $1 m/s$. We generate 50 random sets of network topology for each of the cases with 50, 60, 70, 80, 90, and 100 sensor nodes, respectively.

Our simulation, running on a $2.40 GHz$ CPU, again verifies the efficiency of Concorde. Concorde can obtain the optimal TSP tour for more than 90% of the 50×6 topology sets in 1 second, and all of them are finished in 3 seconds.

A. Tour Length

The css scheme outperforms the Label-Covering algorithm in terms of the resultant tour length noticeably, as shown in Fig. 2 (where the communication distance is $20 m$) and Fig. 3 (where the number of sensor nodes is 50), where TSP represents the length of the optimal TSP tour, COM is the tour length after the combination operation, CSS and LC represent the tour length obtained after the entire css scheme and the Label-Covering algorithm, respectively, and TSP-LB is the lower bound calculated by (2).

The tour length achieved by our css scheme is 83–89% of that obtained by the Label-Covering algorithm, and is about 1.4 times of the lower bound, which cannot be achieved by any practical approximation algorithms [16].

Another observation from Fig. 3 is that, compared with the Label-Covering algorithm, the advantage of the css scheme is more obvious when the communication distance is relatively short (20–50 m). Notice that an important advantage of exploring ME to collect data from sensor nodes is to deal with sparse or disconnected networks, which means the css scheme is specially suitable for these scenarios.

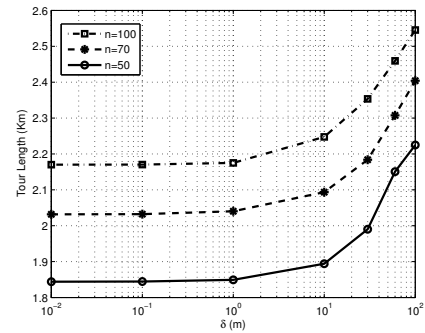


Fig. 4. Effect of δ on the length of resultant traveling tour.

B. The Effect of δ

The control parameter δ in the skip-and-substitute operation directly affects the resultant tour and the computation time. To show its effect, we use a network of 50, 70, and 100 sensor nodes, respectively, with transmission distance of 50 m , and run the simulation with different δ values.

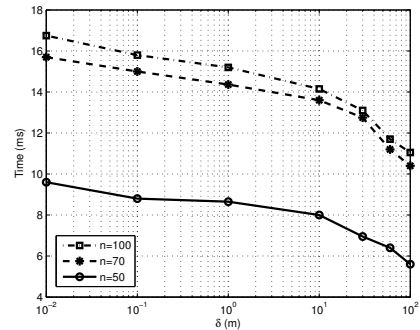


Fig. 5. Effect of δ on the computation time to obtain the traveling tour.

From Fig. 4 and Fig. 5, not surprisingly we can see the tour length reduces as δ decreases, while the computation time increases. However, compared with the reduction in tour length, the increase in computation time is much slower. Further, when δ continues to decrease from 1 m , the tour length reduction comes to a relatively stable stage, so further decrease of δ is not necessary. Changes in both tour length and computation time are sharper when increasing δ continuously from 10 to 100 m , since few segments of the tour are longer than 10 m , and skip-and-substitute can hardly help.

Another observation is that the time spent on the combination operation is much larger than that on skip-and-substitute, where the former is about 5 times higher when $n = 50$ and increases to more than 7 times higher when $n = 100$, agreeing with our complexity analysis in Section IV-C3.

VI. FURTHER DISCUSSION

Many problems on the path selection problem need further exploration, and the css scheme can be extended accordingly. Due to the data rate constraints in wireless communication, a tour intersecting with all the communication disks of sensor nodes may not always be feasible in practice. We have extended the css scheme by considering the data rate constraints but cannot present here due to space limit. The css scheme described above is based on the case of a constant ME speed v . It is worth mentioning that it can also apply to the case of a variable speed $v(t)$: the ME just selects the traveling tour as usual, and moves along it with the maximum speed v_{\max} . MEs can also adjust travel speed to deal with data rate constraints. When there are multiple MEs in the network, normally they are expected to have similar workload and collection latency, thus covering similar sizes of sensing areas and numbers of sensor nodes. If the sensing field is divided into subfields, we can apply the css scheme to each subfield individually at a smaller scale. The collection job can be requested by cluster heads instead of sensor nodes themselves, and cluster heads are rotated among sensor nodes locally to balance energy consumption and MEs are informed along the way. Certainly the ME and the sensing field may have other constraints such as travel trajectory and obstacles. We did not consider them explicitly in this paper, but the progressive optimization approach should apply in these situations as well.

VII. CONCLUSIONS

In this paper, by following the progressive optimization approach, we have proposed a combine-skip-substitute (css) scheme to reduce the tour length, and thus the data collection latency, in wireless sensor networks with mobile elements. We have shown the correctness and complexity of the proposed schemes. Through an extensive simulation study, we have found that the proposed scheme can outperform the best known heuristic algorithm published so far, and the results are within a small range of the lower bound, which cannot be achieved by any practical approximation algorithms. Our future work, in addition to the issues discussed in Section VI, will focus more on extending the css scheme further to the

online scenarios, where the data collection requests arrive at the ME progressively as well.

Acknowledgment: The work is supported in part by the NSERC of Canada, and Liang He is also supported in part by China Scholarship Council and the Ministry of Education's Key Lab for Computer Network and Information Integration at Southeast University, China.

REFERENCES

- [1] X. Xu, J. Luo, and Q. Zhang, "Delay Tolerant Event Collection in Sensor Networks with Mobile Sink," in *Proc IEEE INFOCOM*, 2010.
- [2] Y. Zhuang, J. Pan, and L. Cai, "Minimizing energy consumption with probabilistic distance models in wireless sensor networks," in *Proc IEEE INFOCOM*, 2010.
- [3] S. Panichpapiboon, G. Ferrari, and O. Tonguz, "Optimal transmit power in wireless sensor networks," *IEEE Trans. on Mobile Computing*, vol. 5 no. 10, pp. 1432-1447, Oct. 2006.
- [4] Y. Zhou, Y. Zhang, and Y. Fang, "Access control in wireless sensor networks," *Ad Hoc Networks*, vol 5, no. 1, pp. 3-13, Jan. 2007.
- [5] W. Zeng, R. Sarkar, F. Luo, X. Gu, and J. Gao, "Resilient routing for sensor networks using hyperbolic embedding of universal covering space," in *Proc IEEE INFOCOM*, 2010.
- [6] "NEPTUNE Canada," [Online]. Available: <http://www.neptunecanada.ca>
- [7] "Underwater robots track oil and ocean life," [Online]. Available: <http://spectrum.ieee.org/podcast/robotics/industrial-robots/underwater-robots-track-oil-and-ocean-life>
- [8] M. Todd, et al., "A different approach to sensor networking for SHM: Remote powering and interrogation with unmanned aerial vehicles," in *Proc 6th Int'l Workshop on Structural Health Monitoring*, 2007.
- [9] R. Sugihara and R. Gupta, "Optimizing energy-latency trade-off in sensor networks with controlled mobility," in *IEEE INFOCOM*, 2009.
- [10] R. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs: modeling a three-tier architecture for sparse sensor networks," in *Proc IEEE SNPA Workshop*, 2003.
- [11] A. Chakrabarti, A. Sabharwal, and B. Aazhang, "Using predictable observer mobility for power efficient design of sensor networks," in *Proc ACM/IEEE IPSN*, vol. 2634, pp. 552-568, 2003.
- [12] B. Yuan, M. Orłowska, and S. Sadiq, "On the optimal robot routing problem in wireless sensor networks," *IEEE Trans. on Knowledge and Data Engineering*, vol. 19, no 9, pp. 1252-1261, 2007.
- [13] O. Tekdas, V. Isler, J. Lim, and A. Terzis, "Using mobile robots to harvest data from sensor fields," *IEEE Wireless Communications*, vol. 16, no. 1, pp. 22-28, Feb. 2009.
- [14] M. Ma, and Y. Yang, "SenCar: an energy-efficient data gathering mechanism for large-scale multihop sensor networks," *IEEE Trans on Parallel and Distributed Syst.*, vol. 18, no. 10, pp. 1476-1488, Oct. 2007.
- [15] R. Sugihara, and R. Gupta, "Speed control and scheduling of data mules in sensor networks," in *ACM Trans on Sensor Networks*, to appear.
- [16] S. Safra, and O. Schwartz, "On the complexity of approximating TSP with neighborhoods and related problems," *Computational Complexity*, vol. 14, no. 4, pp. 281-307, 2006.
- [17] A. Dumitrescu, and J. Mitchell, "Approximation algorithms for TSP with Neighborhoods in the Plane," *Journal of Algorithms*, vol. 48, no. 1, pp. 135-159, 2003.
- [18] W. Zhao, and M. Ammar, "Message ferrying: proactive routing in highly-partitioned wireless ad hoc networks," in *IEEE FTDCS*, 2003.
- [19] A. Somasundara, A. Ramamoorthy, and M. Srivastava, "Mobile Element Scheduling with Dynamic Deadlines," in *IEEE Transactions on Mobile Computing*, vol. 6, no. 4, pp. 395-410, April 2007.
- [20] O. Chipara, et al., "Real-time power-aware routing in sensor networks," in *Proc IEEE IWQoS*, 2006.
- [21] "Concorde TSP solver", [Online]. Available: <http://www.tsp.gatech.edu/concorde.html>
- [22] D. Applegate, R. Bixby, V. Chvatal and W. Cook, "On the solution of traveling salesman problems," *Doc. Math. J. DMV Extra*, vol. ICM-III, pp. 645-656, 1998.
- [23] Emo Welzl, "Smallest Enclosing Disks (balls and Ellipsoids)," *Results and New Trends in Computer Science*, vol. 555, pp. 359-370, 1991.
- [24] "Travelling Salesman Problem," [Online]. Available: http://en.wikipedia.org/wiki/Travelling_salesman_problem
- [25] "PowerBot," [Online]. Available: <http://robots.mobilerobots.com>