Dynamic Multi-Channel Packet Scheduling in an Underwater Acoustic Sensor Network

Hamid Ramezani
Faculty of Electrical Engineering,
Delft University of Technology
Email: h.mashhadiramezani@tudelft.nl

Geert Leus
Faculty of Electrical Engineering,
Delft University of Technology
Email: g.j.t.leus@tudelft.nl

***Abstract*—This article considers the broadcasting task in an underwater acoustic sensor network when a few sensor nodes want to transmit their packets to the nodes within their communication range in a collision-free manner. Here, the information about sensor nodes' position and their maximum communication range has been used to minimize the broadcasting time. We have shown that the concept of dynamic channel splitting (multi-channel communications with variable number of channels) has a great impact on the reduction of the broadcasting time. Furthermore, it is shown that in a single hop network, the broadcasting time minimization can be modeled as the multi-depot multiple traveling salesman problem which can be solved sub-optimally through many optimization tools such as Genetic algorithm, and other heuristic methods.**

I. Introduction

Nowadays, due to the numerous marine applications such as seismic monitoring (e.g., for oil extraction), early warning systems (e.g., for tsunamis), military surveillance [1], it is inevitable to utilize underwater acoustic sensor networks (UASNs). However, the underwater acoustic communications is very challenging, and it suffers from low and distance dependent data rate, high power consumption in transmit and receive mode, and long propagation delay. These characteristics have to be considered in different layers of the network such as physical layer and the medium access control (MAC) layer which is responsible for packet exchanges among the nodes. For instance, in a time division multiple access (TDMA) system, in order to decrease the collision probability, the slotted floor acquisition multiple access (FAMA) [2] sets the time slot duration equal to the packet length plus the maximum network propagation delay. However, the acoustic sound speed in an underwater medium is very low (approximately 1500 ms) which causes large propagation delays between the sensor nodes (2 or 3 seconds). The propagation delays are most of the time greater than the packet length (150 to 500 ms), and that makes TDMA approaches inefficient for UASNs. However, the usage of relative position information of the transmitting nodes in the design of MAC protocol could alleviate this inefficiency [3], [4]. In our previous paper, we have shown that using relative position information in the packet scheduling for broadcasting task has a huge impact on a USAN efficiency specially when the ratio of the packet length to the maximum propagation delay is low [5].

In this paper, the concept of dynamic channel splitting has been used to shorten the broadcasting time even more. In this technique, the system decides how to split the total bandwidth into several independent channels, and how to allocate them to the transmitting nodes optimally to minimize the broadcasting time. It is shown that this approach would highly increase the network efficiency.

The structure of the paper is as follows. Section II explains network and physical model. In Sections III, we formulate the problem of single-channel and dynamic multi-channel broadcasting, and discuss about the algorithms that can be employed to find the optimal or sub-optimal solutions. The superiority of the dynamic multichannel over its single-channel counterparts is shown in Section IV through a numerical example. Finally, we conclude the paper in Section V.

II. Network Model

In this paper, we consider a UASN with large number of sensor nodes where among all these nodes, N of them (active nodes) have packets ready for transmission, and they are going to be broadcast these packets to the sensor nodes which are located in their communication range. It is assumed that the number of bits in each packet ($b_n,\ n \in \{1, 2, \ldots, N\}$), the maximum transmission range (D meters), and position information of the active nodes are known at a fusion center, and there is no information about the positions of the other nodes (silent nodes) in the network and they can be anywhere in the operating area. For instance, it can be imagined that these N nodes succeeded in channel acquisition in a contention period, or they are anchors which transmit localization packets periodically in a network, or they are reporting the sensed data periodically to a moving fusion center (data gathering). Therefore, it is reasonable to assume that the information about the positions of the active nodes is available while that of silent nodes is not.

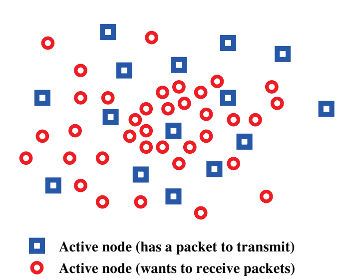

Fig. 1. Network model including active and silent nodes, top view

The goal of the network is to minimize the duration of the broadcasting (or localization) task which is defined as the time of the collision-free transmission of all packets from the transmitting nodes.

A. *Physical layer model*

Developments in software-defined radio systems makes sensor nodes capable of adaptively adjusting their transmission data rate, bandwidth, frequency of operation and so on. Based on this capability, node k is able to transmit its data using the full system bandwidth (or other system resources), with data rate R, i.e., $R_k = R$. Alternatively, the full bandwidth can be split into M sub-bands in a way that the m-th channel ($m \in \{1, 2, ..., M\}$) has data-rate

$$R^{(m)} = \beta^{(m)}(1 - \alpha^{(1:M)})R, \text{ s.t. } \sum_{m=1}^{M} \beta^{(m)} = 1, \quad (1)$$

where $\beta^{(m)}$ is the percentage of the available data rate that a node uses to transmit its packet, $R^{(1:M)} = (1 - \alpha^{(1:M)})R$ is the sum of the data rates of all generated channels after channel splitting, and $\alpha^{(1:M)}$ is the percentage of throughput reduction due to the increase in the number of channels from 1 to M. After channel splitting, node k can then decide to choose the m-th channel and transmit at data rate $R^{(m)}$, i.e., $R_k = R^{(m)}$.

Furthermore, each node at the receiver side is able to detect the transmitted packets regardless of how many channels are generated after channel splitting and which channels the packets are transmitted in.

B. *Collision-free packet transmission*

Since the silent nodes can be located anywhere in the operating area, collisions may occur if the transmitted packets from two or more nodes sweep a point in this area at the same time. Imagine that two transmitting nodes, namely i and j, with respective waiting times w_i and w_j are going to transmit their packets to the nodes in their communication range, and have to satisfy the following conditions

$$w_j > w_{j,\min}, \; w_i > w_{i,\min}, \; w_{j,\min} > w_{i,\min}, \quad (2)$$

where $w_i > w_{i,\min}$ indicates that node i has to start its packet transmission not sooner than $w_{i,\min}$.
Clearly, when the distance between these two nodes is greater than $2D$, their packets never collide with each other. However, when the distance between the two nodes is smaller than $2D$ (defined as connected or collision-risk nodes), the possibility of a collision exists, but we can eliminate this possibility by adjusting the waiting times of the nodes according to the following formulas [5]:

$$w_j > w_i + \frac{b_i}{R_i} + \frac{\tilde{d_{ij}}}{c}, \text{ if } d_{ij} < 2D. \quad (3)$$

$$\tilde{d_{ij}} = \min(d_{ij}, 2D - d_{ij}), \quad (4)$$

where c is the sound speed, and $\tilde{d_{ij}}$ can be considered as a modified distance between two collision-risk nodes (i.e., $d_{ij} < 2D$).
Note that, for simplicity we assumed that the transmitted packet has no effect on the other packets beyond its communication range. However, in reality the interference range has to be considered, and this can be included if a guard ring is added to the maximum transmission range.

A network in which the distances between the connected nodes (collision-risk nodes) are modified (see (4)) is called the *modified network*. With this definition, (3) indicates that to eliminate the possibility of a collision between two collision-risk nodes in a modified network, one node ought to wait until the complete reception of the transmitted packet from the other node before it can start its transmission. The next section briefly reviews how the minimum broadcasting time using a single channel in a network can be obtained [5].

III. Packet scheduling

A. *Single channel Scenario*

In our previous work [5] we have shown that the optimal solution (or solutions) which minimizes the collision-free broadcasting time belongs to at most $N!$ permutations of the transmitting nodes' indexes. In each sequence or permutation, the node whose index appears earlier has to transmit its packet sooner than the ones whose appear later. Conditioned on a given sequence, the minimum duration of the broadcasting task, $T_{\text{broadcast}}$, can simply be computed based on (3). In this procedure, the first transmitting node is assigned to transmit its packet first, then the limitations on the waiting-times of the other nodes are updated. For instance, suppose that in a given sequence the index of node j appears after that of node i, and the mutual distance between them is $d_{ij} < D$. It is again assumed that the conditions in (2) holds here. Hence, node i transmits its packet with data rate $R_i = R$, and after the complete reception of the packet at node j, node j is allowed to broadcast its packet with data rate $R_j = R$ according to its waiting time limit. Under this condition the duration of the collision-free broadcasting from these two nodes can be computed as

$$t_{ij} = w_j + \frac{b_j}{R_j}, \text{ s.t. } w_j > \max(w_{j,\min}, w_i + \frac{d_{ij}}{c} + \frac{b_i}{R_i}), \quad (5)$$

where $\frac{b_i}{R_i}$ is the duration of the packet transmitted by node i. The minimum value of t_{ij} can be obtained if we set

$$w_i = w_{i,\min} \quad (6a)$$

$$w_j = \max(w_{j,\min}, w_{i,\min} + \frac{b_i}{R_i} + \frac{d_{ij}}{c}). \quad (6b)$$

It can be deduced that after scheduling a node (e.g., node i), the remaining nodes (e.g., node j) have to update their waiting times, $w_{j,\min} = w_j$, based on (6). Doing this for each node in the given sequence leads to the minimum value of broadcasting time. Therefore, by comparing the broadcast time of all possible sequences ($N!$ possible sequences)[1], the sequence which leads to the lowest value gives the optimal solution.

In the special case that each transmitting node is a collision-risk neighbor of the other nodes (here referred to as a *fully-connected* network), the nodes have to transmit their packets

[1]Note that, when packet transmission from all transmitting nodes repeated periodically then the number of possible solutions is $(N-1)!$ (Hamiltonian cycle in a complete graph). On the other hand, when transmission of all the packets happens once, there are $N!$ possible solutions for the problem (Hamiltonian path in a complete graph).

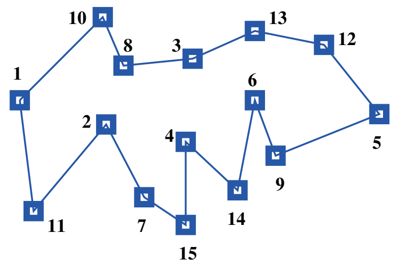

Fig. 2. An example of the Hamiltonian cycle in a network with 15 nodes. The length of the cycle is proportional to the sum of all the propagation delays between transmitting nodes.

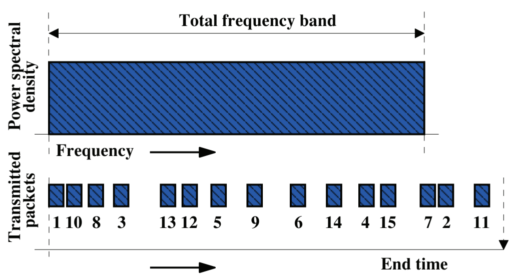

Fig. 3. Sequence of packet transmission in time domain using whole operating bandwidth (single channel scenario).

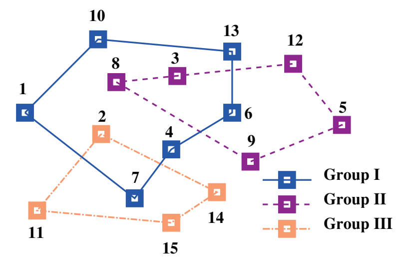

Fig. 4. An example of three groups of nodes which are working at different frequency bands (channels).

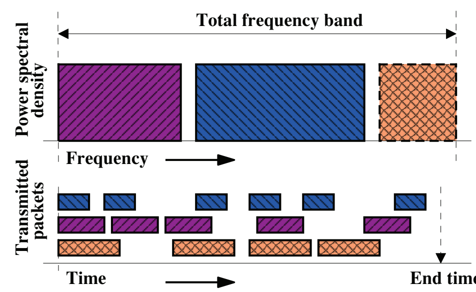

Fig. 5. An example of the parallel transmissions of the packets in time domain from different groups of nodes. The bandwidth allocation in the frequency domain for each group is shown in the upper part of the figure.

one after the other, and the duration of the transmission task is minimized if the Hamiltonian path (or cycle in a periodic transmission) based on the modified distances is formed and the nodes transmit their packets according to this path as depicted in Fig. 2. Finding the Hamiltonian path (cycle) in a network is equivalent to the traveling salesman problem (TSP) which is NP-hard. However, it has been studied extensively [6], and there are many sub-optimum algorithms for that.

Under this condition the broadcasting time can be calculated as

$$T_c = \sum_{n=1}^{N} \frac{b_n}{R} + \frac{d_n}{c} = \frac{\bar{b}}{R} + \frac{\bar{d}}{c}, \tag{7}$$

where $\bar{d}$ is the length of the Hamiltonian path (or cycle) of the modified network, and $\bar{b}$ is the sum of the all number of bits which are transmitted by all nodes at full data-rate, R. In a TDMA system the broadcasting time can be calculated as

$$T_c = \frac{\bar{b}}{R} + N\frac{D}{c}, \tag{8}$$

where $\frac{D}{c}$ is the guard time added to the packet in order to avoid any collision. It is obvious that the position-aware single channel scheduling always leads to a less broadcasting time than that of TDMA approaches.

B. Multi-channel Scenario

Form the previous sub-section it can be seen that in a fully connected network the propagation delays between adjacent nodes play a significant role in the duration of the collision-free broadcasting task. In this sub-section we analyze the minimization of the broadcasting time when several channels in frequency domain with variable bandwidths exist. It is assumed that in each channel a group of nodes can transmit their packets independent of the other nodes which are transmitting in different frequency bands. Note that here the whole bandwidth is divided into M smaller bandwidths, and therefore in comparison with single channel scenario, the length of the packet in each sub-channel increases according to the bandwidth of that sub-channel.

An example of the multichannel transmission with different amount of bandwidth has been shown in Fig.4 and Fig.5. In these figures three groups of transmitting nodes (number of channel is also a variable and can be determined in the fusion center.) are considered and the packet length of each group (with the same number of bits in each packet) vary according to the amount of the bandwidth which has been allocated to each group. Note that there is a small gap between two adjacent channels (deficiency arises in channel splitting technique) which makes the overall data rate smaller. By using this small gap between the adjacent channels, the silent nodes can find out the number of channels and can blindly detect the frequency bands and the packets.

According to the above explanations, the optimization function of minimizing the broadcasting time in a fully-connected network can be formulated as

$$T_c = \min_{l_m,\beta_m} \max \begin{pmatrix} \sum_{l=1}^{l_1} \frac{b_{l,1}}{R^{(1)}} + \frac{d_{l,1}}{c}, \\ \sum_{l=1}^{l_2} \frac{b_{l,2}}{R^{(2)}} + \frac{d_{l,2}}{c}, \\ \dots, \\ \sum_{l=1}^{l_M} \frac{b_{l,M}}{R^{(M)}} + \frac{d_{l,M}}{c} \end{pmatrix} \tag{9}$$

subject to,

$$R^{(m)} = \beta^{(m)}(1-\alpha^{(1:M)})R$$
$$\sum_{m=1}^{M} l_m = N, \text{ and } , 0 \le l_m \le N$$
$$\sum_{m=1}^{M} \beta^{(m)} = 1, \text{ and } \beta^{(m)} > 0$$

where M is the maximum number of channels, $\alpha^{(1:M)}$ determines the ratio of the data-rate loss due to the channel splitting, $\beta^{(m)}$ shows how much bandwidth (from the total remained bandwidth) is allocated to m-th channel, l_m is the number of nodes in each group ($0 \le l_m \le N$), and $(.)_{l,m}$ is the index of the l-th node which is located in the m-th group. Here, the indices of the nodes in each group and the amount of allocated bandwidth for each channel are variables.

The optimization problem in (9) is NP-hard and cannot be solved in polynomial time. Exhaustive search can be employed to find the optimal solution of the problem. In order to do that we have to analyze all possible solutions, find the minimum broadcasting time for each of them, compare their results with each other, and select the one which leads to the minimum value.
Assume that we have one realization of a possible solution as depicted in Fig. 6. based on the order of indices of the nodes in each group, the length of the cycle which connects the nodes can be calculated. Therefore, for this realization the indexes of the nodes in each group and the values of $d_{l,m}$ are known, and only bandwidth allocation has to be accomplished. If $M = 2$ then the optimal bandwidth allocation can easily be obtained through the following formulation

$$\begin{aligned} &\frac{\sum_{l=1}^{l_1} b_{l,1}}{R^{(1)}} - \frac{\sum_{l=1}^{l_2} b_{l,2}}{R^{(2)}} = \frac{\bar{d}_2}{c} - \frac{\bar{d}_1}{c} \\ &R^{(1)} + R^{(2)} = (1-\alpha^{(1:2)})R, \end{aligned} \tag{10}$$

however for the values of $M > 2$, this problem is not convex and has to be solved numerically. All in all, there are at most $\frac{(MN)!}{M(MN-N)!}$ possible realizations[2] which makes finding the optimal solution untraceable.

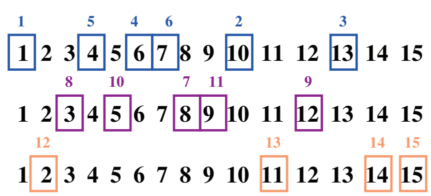

Fig. 6. One realization of the possible solution using at most three sub-channels. The cycle for each group is shown in Fig. 4.

Now, we consider a scenario that each group is allocated the same amount of bandwidth ($\beta_m = \frac{1}{M}$), and each node has a packet with equal number of bits ($b_i = b$) for transmission. Considering each transmitting nodes as vertex of a graph, and defining $v_{ij} = \frac{d_{ij}}{c} + \frac{Mb}{(1-\alpha^{(1:M)})R}$ as the edge weight for the edge between node i and j, the optimization function in (9) would be equivalent to the *multiple traveling salesman problem* (mTSP).

The mTSP is a generalization of the Traveling Salesman Problem (TSP) in which more than one salesman is allowed. Usually, the objective of the mTSP is to determine a tour for each salesman such that the total tour cost is minimized and each city is visited exactly once by only one salesman [7]. However, in the optimization problem we are dealing with, the objective is to minimize the maximum tour length which makes traditional mTSP a little bit different from our problem. In addition in traditional mTSP, all the salesmen start from the same city and come back to that city, but in our problem the salesmen are allowed to depart from any city (approximately similar to multi-depot mTSP [8]). These differences can be included in any algorithm that may be used to solve mTSP problem.

In general, many optimization algorithms have been introduced to tackle the mTSP problem. Linear programming [9], ant colony [10] optimization, genetic algorithm (GA) and other meta-heuristic methods are examples of these algorithms. Among them we have chosen a simple form of GA [11]. Basically, GA works by generating a population of possible solutions (called chromosomes) to a problem. Usually in the process of the algorithm chromosomes are created by operators such as *crossover*, *mutation*, and *elitism*. The crossover operator exchanges the elements between the groups, the mutation operators replaces the elements of a vector (ordering in a group) randomly, and elitism is the process of selecting the better possible solution or with a bias towards the better ones. After, the generated chromosomes are evaluated, the ones which fit best survive and the rest are eliminated. This process continues until a pre-specified good solution is found, an amount of computing time passes, or until no significant improvement occurs in the population for a given number of iterations [11].

IV. Simulations

For the simulation part, we consider a fully connected network with 12 transmitting nodes where they are located on the surface of an square shaped operating area as illustrated in Fig. 7. All the nodes have equal number of bits for transmission, and it is assumed that the generated channels have equal bandwidths too. Furthermore, the frequency gap between two adjacent channels is set to 5% of the total data rate, i.e., $\alpha^{1:M} = 0.05(M-1)$. In order to find the suboptimal solution, we use the GA with 10^4 iterations, and population size of 80.

In Fig. 8, we have shown the effect of dynamic multi-channel scheduling for different values of packet lengths. The packet length at full data-rate, R, varies from $\frac{1}{5}$ to 1 second. It can be observed that there is an optimal value for the number of channels that minimizes the broadcasting time. This depends on the number of bits in each packet, the configuration of the transmitting nodes, and the value of data-rate deficiency, $\alpha^{(1:M)}$.

When the packet length in comparison with the maximum propagation delay (here $\frac{D}{c} = 3\sqrt{2}$) is small, the sum of the propagation delays between transmitting nodes plays the main

[2]The M in the denominator indicates that there is no difference between the sub-channels, and we do not need to analyze all N permutation of NM. See also footnote 1 which explains the difference between the complexity of finding optimum cycle and path.

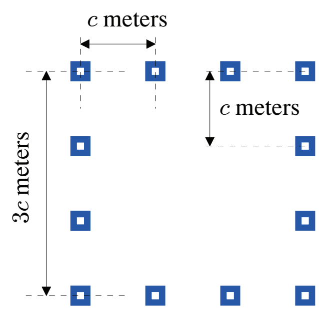

Fig. 7. The positions of the 16 transmitting nodes on the surface of the operating area.

role in the value of the broadcasting task. On the other hand, as the number of (sub) channels gets larger, the length of the packets increases, and in contrary the sum of the propagation delays in each group gets smaller because each group works independently with less number of transmitting nodes and in a smaller area which makes the overall performance better. Moreover, we can also observe that the gain we obtain by using the dynamic multichannel allocation is not very much when the ratio of packet length (using whole bandwidth) to the maximum propagation delay is large. However, when this ratio is small the efficiency can be as big as 100% to 200%.

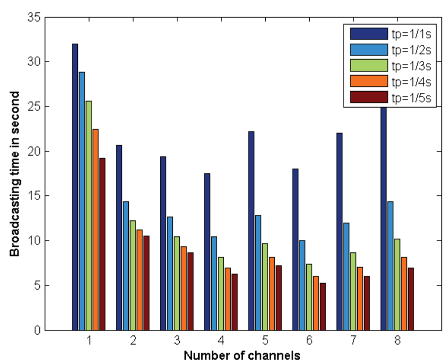

Fig. 8. The effect of the number of channels on the broadcasting time. In the algorithm, the best number of channels is selected for the broadcasting task.

V. Conclusion

We have considered the problem of packet scheduling in an Underwater acoustic sensor network where a few nodes are going to broadcast their packets in a collision-free manner. The application of this problem could be localization, and data gathering. We mentioned that the position information of the transmitting nodes can be utilized to reduce the duration of the broadcasting task. In addition, we have proposed a dynamic multi-channel (DMC) allocation which adaptively adjusts the number of channels to minimize the broadcasting time. Having formulated the problem, we show that the minimization function in NP-hard, but can be modeled by the famous multiple traveling salesman problem. In this regard, using a simple genetic algorithm we show that the DMC-based algorithms outperform the single channel ones greatly.

Acknowledgment

The research leading to these results has received funding from the European Commission FP7-ICT Cognitive Systems, Interaction, and Robotics under the contract #270180 (NOPTILUS).

References

[1] J. Heidemann, Wei Ye, J. Wills, A. Syed, and Yuan Li, "Research challenges and applications for underwater sensor networking," in *Proc. IEEE Wireless Communications and Networking Conf. WCNC 2006*, 2006, vol. 1, pp. 228–235.

[2] M. Molins and M. Stojanovic, "Slotted FAMA: a MAC protocol for underwater acoustic networks," in *Proc. OCEANS 2006 - Asia Pacific*, 2006, pp. 1–7.

[3] Lu Hong, Feng Hong, Zhongwen Guo, and Xiaohui Yang, "A TDMA-based MAC protocol in underwater sensor networks," in *Proc. 4th Int. Conf. Wireless Communications, Networking and Mobile Computing WiCOM '08*, 2008, pp. 1–4.

[4] Yang Guan, Chien-Chung Shen, and J. Yackoski, "MAC scheduling for high throughput underwater acoustic networks," in *Proc. IEEE Wireless Communications and Networking Conf. (WCNC)*, 2011, pp. 197–202.

[5] Hamid Ramezani and Geert Leus, "L-MAC: Localization packet scheduling for an underwater acoustic sensor network," *Accepted in Proc. of the International Conference on Communications, ICC2013*, 2013.

[6] Vasek Chvtal & William J. Cook David L. Applegate, Robert E. Bixby, *The Traveling Salesman Problem: A Computational Study*, Princeton University Press. ISBN 978-0-691-12993-8., 2006.

[7] Tolga Bektas, "The multiple traveling salesman problem: an overview of formulations and solution procedures," *Omega*, vol. 34, no. 3, pp. 209 – 219, 2006.

[8] Enrique Benavent and Antonio Martínez, "Multi-depot multiple tsp: a polyhedral study and computational results," *Annals OR*, vol. 207, no. 1, pp. 7–25, 2013.

[9] Imdat Kara and Tolga Bektas, "Integer linear programming formulations of multiple salesman problems and its variations," *European Journal of Operational Research*, vol. 174, no. 3, pp. 1449 – 1458, 2006.

[10] Soheil Ghafurian and Nikbakhsh Javadian, "An ant colony algorithm for solving fixed destination multi-depot multiple traveling salesmen problems," *Appl. Soft Comput.*, vol. 11, no. 1, pp. 1256–1262, 2011.

[11] A. E. Carter, *Design and application of genetic algorithms for the multiple traveling salesperson assignment problem*, Ph.D. thesis, Virginia Tech, USA, 2003.