Multi-training sensor networks with bipartite conflict graphs

Ruzana Ishak
Department of Mathematics
College of Science and Technology
Universiti Teknologi Malaysia
54100 Kuala Lumpur, Malaysia
ruzanaishak@yahoo.com

Stephan Olariu
Department of Computer Science
Old Dominion University
Norfolk, VA 23529-0162,USA
olariu@cs.odu.edu

Shaharuddin Salleh
Department of Mathematics
Faculty of Science
Universiti Teknologi Malaysia
81310 Johor Bahru, Malaysia
ss@mel.fs.utm.my

ABSTRACT

Due to their potential applications in various situations such as battlefield communications, emergency relief, environmental monitoring, and other special-purpose operations, wireless sensor networks have recently emerged as a new and exciting research area that has attracted a good deal of well-deserved attention in the literature. In this work we take the view that a sensor network consists of a set of tiny sensors, massively deployed over a geographical area. The sensors are capable of performing processing, sensing and communicating with each other by radio links. Alongside, with the tiny sensors, more powerful devices referred as *Aggregating and Forwarding Nodes*, (AFN, for short) are also deployed. In support of their mission, the AFNs are endowed with a special radio interface for long distance communications, miniaturized GPS, and appropriate networking tools for data collection and aggregation. As a fundamental prerequisite for self-organization, the sensors need to acquire some form of location awareness. Since fine-grain location awareness usually assumes that the sensors are GPS-enabled, in the case of tiny sensors the best we can hope for is to endow them with coarse-grain location awareness. This task is referred to as *training* and its responsibility lies with the AFNs. However, due to the random deployment, some of the sensors fall under the coverage area of several AFNs, in which case the goal is for these sensors to acquire location information relative to all the covering AFNs. The corresponding task is referred to as *multi-training*.

The main contribution of this work is to show that in case the conflict graphs of the AFN coverage is *bipartite*, multi-training can be completed very fast by a simple algorithm.

Categories and Subject Descriptors

C.2.0 [**Computer-communication Networks**]: Wireless Sensor Networks; C.2.2 [**Network Protocols**]: Localization Protocols

General Terms

Algorithms, Design, Performance

Keywords

Aggregating and Forwarding Nodes, multi-training, bipartite

1. INTRODUCTION

As the technology for wireless communications advances and the cost of manufacturing a sensor node continues to decrease, a low-cost but yet powerful sensor network may be deployed for various applications that can be envisioned for daily life. Integrating simple processing, storage, sensing, and communication capabilities into small-scale, low-cost devices and joining them into so-called wireless sensor networks opens the door to a plethora of new applications[1, 7, 8]. A wireless sensor network consists of a possibly large number of wireless devices which able to record environmental measurements such as temperature, light, sound and humidity. The sensor readings are transmitted over a wireless channel to a running application that makes decisions based on these sensor readings. By integrating computation and control in our physical environment, the well-known interaction paradigms of person-to-person, person-to-machine and machine-to-machine can be supplemented, in the end, by a notion of person-to-physical world; the interaction with the physical world becomes more important than mere symbolic data manipulation[3, 4, 7, 5].

Some applications benefit or even require that the sensory data collected by sensors be suplemented with location information, which encourages the development of communication protocols that are location aware and perhaps location dependent[1, 2, 11]. The practical deployment of many sensor networks will result in sensors initially being unaware of their location: they must acquire this information post-deployment. In fact, in most of the existing literature, the sensors are assumed to have learned their geographic position. The *location-awareness* problem is for individual sensors to acquire location information either in absolute form (e.g.,geographic coordinates) or relative to reference points.

The localization problem is for individual sensors to determine, as precisely as possible, their geographic coordinates in the area of deployment.

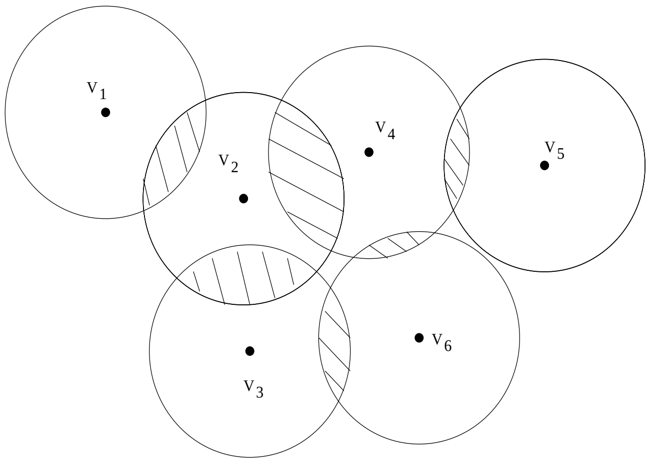

Figure 1: Illustrating the multi-training.

Exact geographic location is not necessary in some applications, all that individual sensors need is *coarse-grain* location awareness. One can obtain this course-grain location awareness by a *training* protocol that imposes a coordinate system on the sensor network. Olariu et al. [9] and Wadaa et al. [12] proposed an interesting training protocol that provides partitioning into clusters and a structured topology with natural communication paths. But, they only discussed for one sink. In this paper, we will show the result for several sinks in training the sensors. Such a scenario, involving six sinks v_1, v_2, v_3, v_4, v_5 and v_6 is depicted in Figure 1.

Alongside, with the tiny sensors, more powerful devices referred as *Aggregating and Forwarding Nodes*, (AFN, for short) are also deployed. As their names suggests, the AFNs aggregate the sensory results collected by the sensors in their neighborhood and forward the suitably aggregated information to an end user that may be collocated with the sensors or remote. Training the sensor nodes with a *one AFN* and *two AFNs* are already a challenging task. In this paper, training *multiple AFNs* has additional complexities. We address the task of training where some sensors must be trained by several AFNs whose coverage areas overlap. The *training* can be performed by a protocol that is at the same time lightweight and secure as mentioned in [12]. The authors show that in case the conflict graphs of the AFN coverage is *bipartite*, multi-training can be completed very fast by a simple algorithm.

The remainder of this paper is organized as follows. Section 2 discusses the background of sensor network training used throughout the work. Section 3 and 4 presenting the theoretical of the algorithm for the protocol in performing the networks as a conflict graph. Section 5 discusses the problem for other type of conflict graphs. Finally, Section 6 offers concluding remarks.

2. COORDINATE SYSTEM

The task of training refers to imposing a coordinate system onto the sensor network in such a way that each sensor belongs to exactly one sector. The coordinate sector divide the sensor network area into equiangular wedges. In turn, these wedges are divide into sectors by means of concentric circles or coronas centered at the sink and whose radii are determined to optimize the transmission efficiency of sensors-to-sink. Referring to Figure 2, the task of training a sensor network involves establishing:

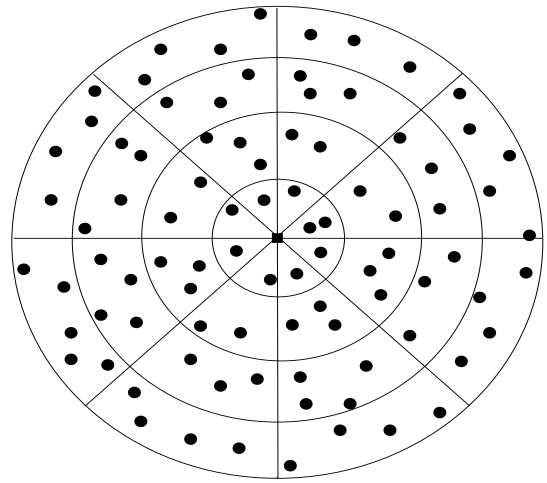

Figure 2: A trained sensor network.

Coronas: The deployment area is covered by k coronas determined by k concentric circles of radii $0 < r_1 < r_2 < \cdots < r_k = R$ centered at the sink.

Wedges: The deployment area is ruled into a number of angular wedges centered at the sink. Wedges are established by directional transmission [9].

As illustrated in Figure 2, at the end of the training period each sensor has acquired two coordinates: the identity of the corona in which it lies, as well as the identity of the wedge to which it belongs. Importantly, the locus of all the sensors that have the same coordinates determines a cluster.

3. CONFLICT GRAPH

R.Ishak *et al.* [6] have discussed and proved that the total time training for two $AFNs$ (dual-training) is $k + 2^{\alpha} - 2$ where $2^{\alpha-1} < q \leq 2^{\alpha}$ for $q(1 \leq q \leq k)$ corresponding disks overlap.

In this section, we consider the case where several AFN_s are deployed in an area of interest as shown in Figure 3. Visibly, some of the disks surrounding these AFN_s are overlapping. The question that we address is to determine the overall training time of the sensors in the union of the corresponding disks.

More formally, let AFN_1, AFN_2, $\ldots$, AFN_n $(n \geq 2)$ be deployed in an area and let D_1,D_2, $\ldots$, D_n be the corresponding disks, such that for every choice of i, j $(1 \leq i < j \leq n)$ D_i and D_j overlap in q_{ij} coronas. Notice that some of the $q'_{ij}s$ may be zero. As an illustration, the matrix below

Table 1: Multi-training matrix

$i\backslash j$	D_1	D_2	D_3	D_4	D_5	D_6
D_1	$---$	q_{12}	0	0	0	0
D_2	$---$	$---$	q_{23}	q_{24}	0	0
D_3	$---$	$---$	$---$	0	0	q_{36}
D_4	$---$	$---$	$---$	$---$	q_{45}	q_{46}
D_5	$---$	$---$	$---$	$---$	$---$	0
D_6	$---$	$---$	$---$	$---$	$---$	$---$

capture the section of disks in Figure 3 (only the non-zero entries are recovered).

Moreover, for every q_{ij}, $(1 \le i < j \le n)$, we let α_{ij} be the natural power for which

$$2^{\alpha_{ij}-1} < q_{ij} \le 2^{\alpha_{ij}} \quad (1)$$

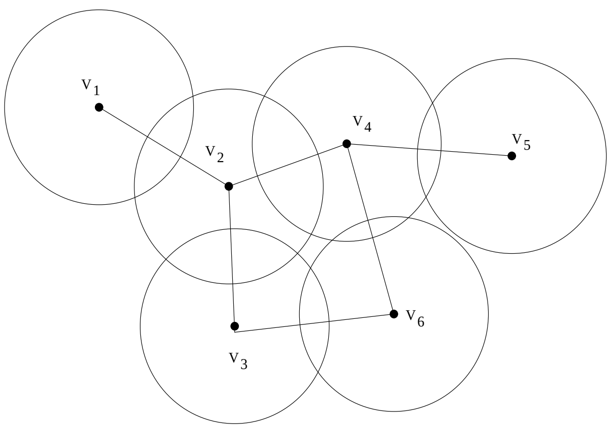

Figure 3: Illustrating a multi-AFNs graph of sensor network.

Let $\alpha = max_{1 \le i<j \le k}\{\alpha_{ij}\}$. We model the problem as follows. Consider the graph $G = (V, E)$ with $V = \{v_1, v_2, \ldots, v_n\}$ such that $\forall i,(1 \le i \le n)$, $v_i \equiv AFN_i$. For example, two vertices v_i and v_j in G as the *conflict* graph of the situation that it is intended to model. The conflict graph of the six AFN_s in Figure 3 is illustrated in Figure 4 below.

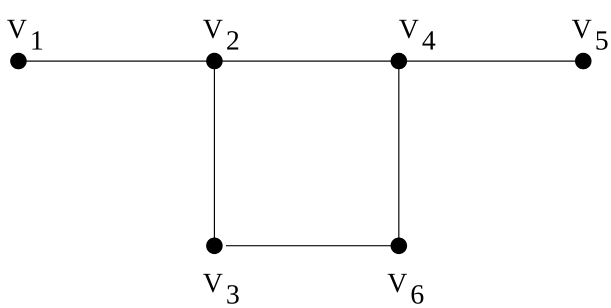

Figure 4: Illustrating the conflict graph of six AFNs.

4. BIPARTITE SELF-ORGANIZATION PROTOCOL

In Multiple training, if AFNs are associated as a graph $G = (V, E)$ and if the conflict graph is bipartite, the total training time is as efficient as for dual-training. To initiate the result, before the AFNs start training the sensor nodes, it should determine its location. Once deployed, the AFNs will reorganize its location by using a protocol to determine whether the graph G is bipartite. The protocol is as follows.

1. Once deployed, the $AFNs$ which have been injected with IDs will send their IDs to its neighbors. Each AFN will receive IDs information from their neighbors. Upon receiving, AFN will only transmit the lowest IDs received to the neighbors. This process will continue until the lowest ID has been circulated to all $AFNs$.

2. The AFN which has the lowest ID chooses a color and transmits it to its neighbors. Every node that receives a color message from a neighbor chooses the appropriate color for itself.

3. As we shall proved shortly, if the graph is bipartite, all the vertices will be colored black or white such that no vertices of the same color are adjacent.

4. When this process has terminated, all the AFN will start training in parallel. In this protocol, we assume that black $AFNs$ have the priority to complete the training at time $k-1$ and white $AFNs$ must wait at a certain time period to compute the training.

In the case of bipartite graph, the total time for training can be completed in time $k + 2^{\alpha} - 2$. The correctness of protocol 3 is implied by the following. Let $G = (V, E)$ be a graph with a distinguish vertex v_0. A two-coloring of G(using colors B and W) is said to be "proper" if every vertex receives exactly one color and no edge in G has end points of the same color. Our goal is to derive a simple protocol to properly color a bipartite graph with two colors as above. The protocol is the following.

1. v_0 selects a color at random, say B, and announces its color to its neighbors by broadcasting in time slot 1.

2. In time slot 2, all the vertices that have heard from v_0 in slot 1 select the complementary color (W, in this case) and announce it to all their neighbors.

3. This is continued until all the vertices have received their colors.

Observe that the protocol described perform s a Breath-First-Search of G anchored at v_0. Therefore, we can think of G as a layered graph as shown in Figure 5. It is to see that for $1 \le i \le t$, $N_i(t)$ is the set of vertices that receive colors in the $(i+1)^{th}$ time slot. We need to show that the above procedure properly colors a connected graph if and only if it is bipartite.

THEOREM 1. *A connected graph G with a distinguished vertex v_0 is bipartite if and only if the previous protocol produces a proper two-coloring of G.*

PROOF. Firstly, if the two-coloring obtained at the end of the protocol is proper than G is certainly bipartite. In fact, the bi-partition is indicated by the individual color sets.

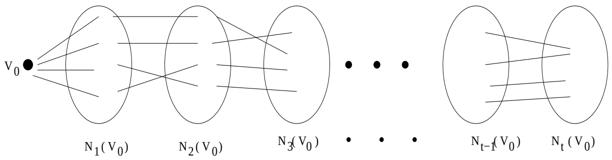

Figure 5: Illustrating the layered graph of G.

Conversely, let G be bipartite. We prove the theorem by induction on t, the largest distance from v_0 to any vertex in the graph.

Basis: If $t = 1$ then the graph consists of a central vertex v_0 as shown in Figure 6. When v_0 transmits its color, all the other vertices will set their color to W. Since the graph is bipartite no vertex in $N_i(v_0)$ are adjacent and consequently the B/W coloring satisfies the condition of the theorem.

Induction Step: Assume that the statement is true for all graph for which the largest distance for the elected vertex v_0 to any other vertex is strictly smaller than t. Now, consider the graph G' obtained from G by removing the vertices in $N_t(v_0)$. We claim that G' is connected. To see this, consider arbitrary vertices u and v in G'. We only need to show that u and v are connected by a path in G' itself. Assume without loss in generality that $u \epsilon N_i(v_0)$ and $v \epsilon N_j(v_0)$. Now, there is a path from u to v that goes first from u down to v_0 and then from v_0 to V. Thus G' is connected.

By the induction hyphotesis, when the protocol is applied to G', a B/W coloring satisfying the properties of the theorem is obtained. In particular, the vertices in $N_{t-1}(v_0)$ are all colored the same. Now, consider what happens when we add back the vertices in $N_t(v_0)$ connecting them to other vertices in G. Note that all these added edges connect vertices from $N_t(v_0)$ with vertices in $N_{t-1}(v_0)$ only. $\square$

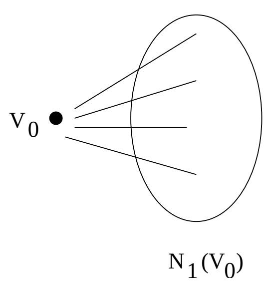

Figure 6: Illustrating the case of t=1 at layered graph.

Consequently, upon vertices in $N_{t-1}(v_0)$ broadcasting their color, the vertices in $N_t(v_0)$ will receive a consistent color. This completes the proof of the theorem.

5. OTHER CONFLICT GRAPHS

As an extension for the work done by R.Ishak [6], we consider the case where $2^\gamma \leq 2^\beta \leq 2^\alpha$ in which the $AFNs$ are numbered 1, 2, 3 as shown in Figure 7. The $AFNs$ are numbered in the order of their priorities. The time line below shows the training progress.

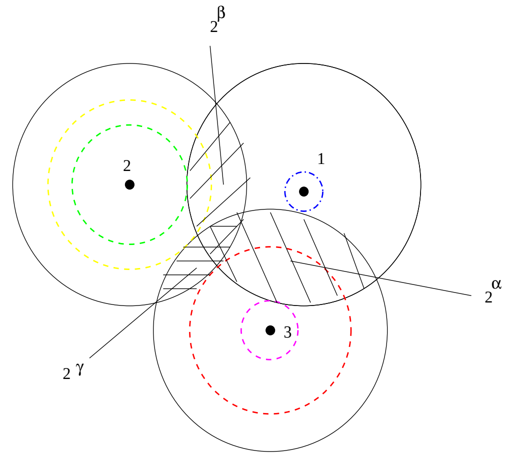

Figure 7: Illustrating the priority multi-training.

- In slot 1, all three $AFNs$ start training in parallel.
- At the end of slot k-2^α, AFN_3 stops training.
- At the end of slot k-2^β, AFN_2 stops training.
- At the end of slots $k-1$, AFN_1 has completed training.
- In slot k, AFN_2 and AFN_3 resumes training.
- In slot $k + 2^\beta - 2$, AFN_2 completes training.

The time line progress for this training has been illustrated in Figure 8.

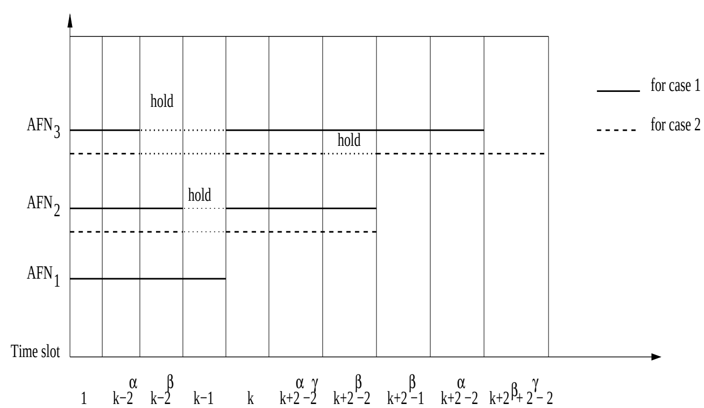

Figure 8: Illustrating time line algorithm.

Since AFN_3 has the lowest priority, it has to defer to AFN_2 if necessary. There are two possibilities to be considered. Refer to Theorem 2

THEOREM 2. *Consider three $AFNs$ with overlap areas of $0 \leq P \leq \varepsilon \leq r \leq k$ coronas and with $2^{\alpha-1} < r \leq 2^\alpha$, $2^{\beta-1} < \varepsilon \leq 2^\beta$ and $2^{\gamma-1} < P \leq 2^\gamma$. Then the training of the sensors covered by at least one of the $AFNs$ can be completed in $k - 2 + Max\{2^\alpha, 2^\beta + 2^\gamma\}$time slots.*

PROOF. 1. If $2^\alpha - 2^\gamma > 2^\beta \Leftrightarrow 2^\alpha > 2^\beta + \beta 2^\alpha$, then clearly AFN_2 completes before AFN_3 has had a chance to get to the conflict zone. Therefore, AFN_3 does not have to defer and will complete its training in slot $k + 2^\alpha - 2$(see Figure 7).

2. If $2^\alpha - 2^\gamma < 2^\beta$,then AFN_3 has to defer to AFN_2. This situation is potrayed in Figure 9. AFN_3 reaches the point A before AFN_2 has enough time to complete training. In this case, AFN_3 must stop in slot $k+2^\alpha - 2^\gamma$ to wait for AFN_2 to complete in slot $k + 2^\beta - 2$. AFN_3 will restart in slot $k + 2^\beta - 1$ and complete training in slot $k + 2^\beta + 2^\gamma - 2$.

□

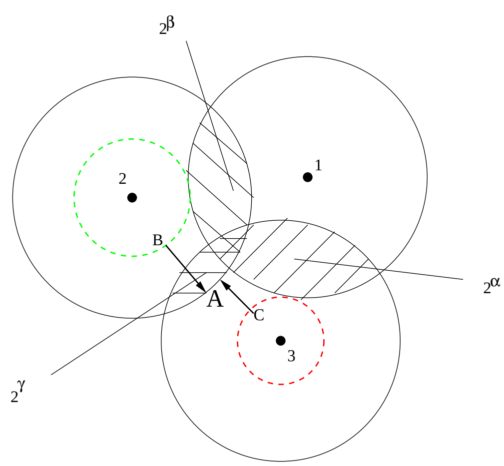

Figure 9: Illustrating the priority multi-training for case 2.

Thus, training was completes in $k - 2 + Max\{2^\alpha, 2^\beta + 2^\gamma\}$ slots.

6. CONCLUDING REMARKS

We have proposed an algorithm for a self organization training for multi sinks referred to as Aggregating and Forwarding Nodes (AFN for short). The training as discussed by [6, 9, 12] is the process of learning the coordinates by the sensor nodes.

In this protocol, by performing a bipartite graph, the AFN can train the sensor nodes in the entire network efficiently. The training for multiple sinks provides a flexible, open framework in which various specific training can be represented and modeled.

7. REFERENCES

[1] F.Akyldiz, W. Su, Y. Sankarasubramanian, and E. Cayirci, Wireless sensor networks: A Survey, *Computer Networks*,38(4),2002,393-422.

[2] N.Bulusu, J.Heidemann, and D.Estrin, GP-less low cost outdoor localization for very small devices, *IEEE Personal Communications*, 7(5): 28-34, 2000.

[3] D. Culler, D. Estrin, and M. Srivastava, Overview of sensor networks, *IEEE Computer*, 37(8), 2004, 41–49.

[4] D. Culler and W. Hong, Wireless sensor networks, *Communications of the ACM*, 47(6), 2004, 30–33.

[5] K. Martinez, J. K. Hart and R. Ong, Environmental sensor networks, *IEEE Computer*, 37(8), 2004, 50-56.

[6] R.Ishak, S.Olariu, S.Salleh and Q.Xu, Dual-training for Massively- Deployed Sensor Networks,*13th International Conference of Telecommunications*, Portugal, May 2006.

[7] Karl,H., and Willig,A. (2005). *Protocols and Architectures for Wireless Sensor Networks*.John Wiley and Sons Ltd, England.

[8] S.Olariu, Q.Xu, A.Wadaa and I.Stojmonovic, A virtual infrastructure for wireless sensor networks, in I. Stojmenivc, Ed., *Handbook of Sensor Networks*, Wiley 2005, 107–140.

[9] S.Olariu, A.Wadaa, L.Wilson and M.Eltoweissy, Wireless sensor networks: leveraging the virtual infrastructure, *IEEE Network*,18(4), 204, 51-56.

[10] S.Olariu, M.Eltoweissy, and M.Younis, ANSWER: Autonomous Wireless Sensor Network, *Proc. ACM Q2SWinet*, Montreal, Canada, October 2005.

[11] K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie, Protocols for self-organization of a wireless sensor network, *IEEE Personal Communications*, October 2000, 16–27.

[12] A.Wadaa, S.Olariu, L.Wilson, M.Eltoweissy, and K.Jones, Training a Wireless Sensor Network, *Mobile Networks and Applications* 10, 151-168, 2005.