ALGORITHMIC MODELS FOR SENSOR NETWORKS

Seminar work of Alexander Grinin

Chair of Informatics 1,
RWTH Aachen, Ahornstr. 55,
52074 Aachen, Germany
Project chairman: Alexander Fanghänel

February 12, 2008

Abstract. This paper briefly shows some algorithmic models for sensor networks and discusses their characteristics. The selection of models presented is of course far from being complete and also highly subjective. The main focus is based on most important models of higher levels of abstraction; a large body of interesting work about the physical layer is not considered.

Keywords: Sensor Networks, Distributed Algorithms, UDG, BIG, SINR.

1 Introduction

1.1 What are sensor networks?

A sensor network is a set of small autonomous systems, called sensor nodes which cooperate to solve at least one common application. Their tasks include some kind of perception of physical parameters [1]. Sensor networks are still in the development stage, so there are no practical applications invented yet. Mostly they serve experimental and demonstration purposes only. However, examples for interesting applications can already be identified. Small sensor nodes could be used for medical applications, e.g. for the surveillance of elderly people. In the field of business administration, stock keeping and surveillance could be revolutionized. The sensors could also be used to monitor temperature and humidity.

1.2 Features of sensor networks

A sensor node is like an ordinary computer with a processor and data storage. It can be equipped with one or more sensors and a module for wireless communications. All parts are powered with a little battery. As a communication medium the radio technology is used. Sensor nodes receive no new external energy reserves, which mean if the battery is down the life of the node is exhausted. Therefore the battery must be as efficiently as possible, while all other parts of a node must have low power consumption. A standby mode e.g. can be a solution. The cost of the hardware should be as low as possible, so that sensor networks could be financially acceptable to install[1].

1.3 Brief description of models

The choice of descriptive model for sensor networks is also very important. In order to develop suitable algorithms for sensor networks and in order to give math-

[1] http://de.wikipedia.org/wiki/Sensornetz

ematical proofs of their correctness and performance, appropriate models are needed. Finding good models however is a challenging task. On the one hand, a model should be as simple as possible such that the analysis of a given algorithm remains tractable. On the other hand, however, a model must not be too simplistic in the sense that it neglects important properties of the network. A great algorithm in theory may be inefficient or even incorrect in practice if the analysis is based on idealistic assumptions. For example, an algorithm which ignores interference may fail in practice since communication happens over a shared medium [2].

2 Connectivity models

2.1 General problems

A first and one of the most important questions in sensor networks is connectivity. How could the information be efficiently transmitted between the nodes and which models describe it best? What conditions should be fulfilled? The following models try to answer these questions.

2.2 Unit disk graph (UDG)

The classic connectivity model from computational geometry is the so-called unit disk graph (UDG). **UDG is a graph $G = (V, E)$ formed from a collection of equal-radius circles, in which two of them are connected by an edge if one containing the center of the other circle**[2]. The Euclidean distance in UDG is normalized to 1. That is, for arbitrary u, $v \in V$, it holds that $\{u, v\} \in E \Leftrightarrow |u, v| \leq 1$ Figure 1 depicts an example of a UDG. The UDG model is quite idealistic: in reality, radios are not omnidirectional, and even small obstacles such as plants could change connectivity.

2.3 Quasi unit disk graph (QUDG)

Another model that is quite similar to UDG is Quasi Unit Disk Graph (QUDG). **The nodes are in arbitrary positions in $\mathbb{R}^2$. All pairs of nodes with Euclidean distance at most ρ for some given $\rho \in (0, 1]$ are adjacent. Pairs with a distance larger than 1 are never in each other's transmission range. Finally, pairs with a distance between ρ and 1 may or may not be neighboring**[2]. An example is shown in Figure 2. Also a QUDG with $\rho = 1$ is a UDG.

[2]http://en.wikipedia.org/wiki/Unit_disk_graph

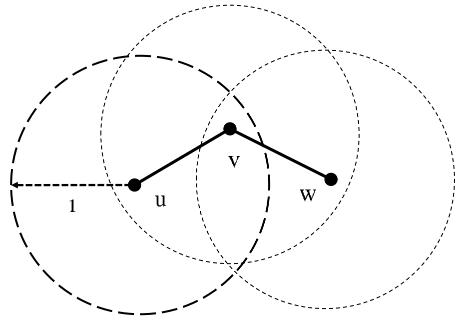

Figure 1: Unit disk graph: node u is adjacent to node v (distance ≤ 1), but not to node w (distance > 1).

2.4 Bounded independence graph (BIG)

But how realistic is QUDG? If there is e.g. a wall u and v can be close but not adjacent. The model of Bounded Independence Graph (BIG) shows us the solution of UDG and QUDG disadvantages. In complex environments such as buildings, the neighbors of a node often have bounded independence. Within the walls connectivity typically adheres to certain geometric constraints. Hence the resort to general connectivity graphs is way too pessimistic.Therefore for any two vertices $u, v \in V$, it could be denoted that $d(u, v)$ is the shortest hop-distance between u and v. Using this, $\Gamma_r(v)$ is called the (closed) r-neighborhood of v. And for any $r \geq 0$, and $v \in V$, it is defined that $\Gamma_r(v) = \{u \in V | d(u, v) \leq r\}$. **An undirected graph $G = (V, E)$ is called BIG if there exists a function $f(r)$ such that for every $v \in V$ and $r \geq 0$, the size of the largest independent set in the r-neighborhood $\Gamma_r(v)$ is at most $f(r)$**[4]. The BIG model reflects reality quite well: the node searches not only for his closest neighbor but for all of them within his radius of connectivity. E.g. if node v cannot find the node u or the node w because of the wall he might try to communicate with it using another two nodes as his agents, see Figure 3. A set of nodes is called independent only if all nodes in the set are pair wise not adjacent. In this case sending and receiving of the signals is not further possible.

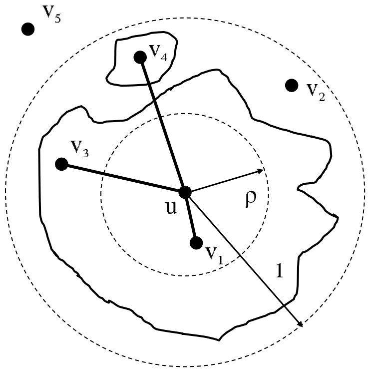

Figure 2: Quasi unit disk graph from the perspective of node u: Node u is always adjacent to node v_1 ($d(u, v_1) \leq \rho$) but never to v_5 ($d(u, v_5) > 1$). All other nodes may or may not be in u's transmission range. In this example, node u is adjacent to v_3 and v_4, but not to v_2.

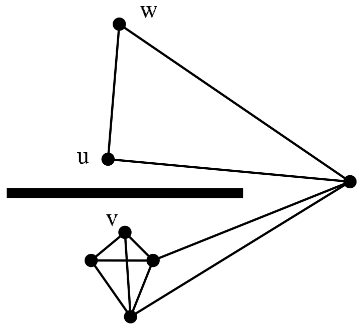

Figure 3: Nodes u and v are separated by a wall. Nodes on the same side of the wall are completely connected. However, due to the wall, although u can reach a distant node w, it cannot hear the close node v. Such situations can be modeled by the BIG, but not by the UDG or the QUDG.

3 Interference models

3.1 General problems

In wireless networks where the communication medium is shared, transmissions are exposed to interference. That means that the information from node u may not reach the node v, because of a concurrent transmission of other nodes going nearby. A signal might for example interfere with itself due to multipath propagation or with another signals being send or received. Therefore interference models with the worst-case perspective capture have to be proposed.

3.2 Signal-to-interference-plus-noise ratio (SINR)

The Signal-to-Interference-plus-Noise Ratio (SINR) is an important metric of wireless communication link quality [3]. It goes a bit further extending UDG and could be described as a relation of the signal power P_r received by a node v_r with the amount of interference I_r generated by other nodes. **A node v_r receives a transmission if and only if $\frac{P_r}{N+I_r} \geq \beta$, where N is an ambient noise power level and β is a small constant (depending on the hardware) that denotes the minimum signal to interference ratio that is required for a message to be successfully received**[2]. The value of the received signal power P_r is a decreasing function of the distance $d(v_s, v_r)$ between transmitter v_s and receiver v_r. More specifically, the received signal power is modeled as decaying with distance $d(v_s, v_r)$ as $\frac{1}{d(v_s,v_r)^\alpha}$. The so-called path-loss exponent α is a constant between 2 and 6 and depends on external conditions of the medium, as well as on the exact sender-receiver distance. Therefore a message transmitted from a node $v_s \in V$ is successfully received by a node v_r if

$$\frac{\frac{P_s}{d(v_s,v_r)^\alpha}}{N + \sum_{v_i \in V\{v_s\}} \frac{P_i}{d(v_s,v_r)^\alpha}} \leq \beta.$$

3.3 UDG with distance interference

In this model nodes are situated arbitrarily in the plane. **Two nodes can communicate because of the constant transmission power directly if and only if their Euclidean distance is at most 1, and if the receiver is not disturbed by a third node with Euclidean distance less or equal a constant $R \geq 1$**[2]. Figure 4 shows an example.

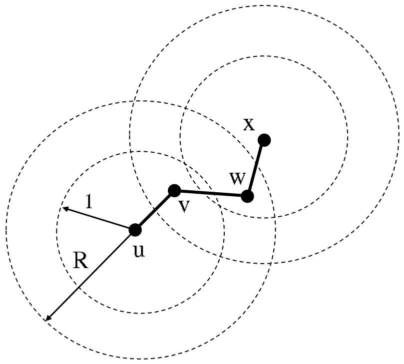

Figure 4: The UDI model has two radii: a transmission radius (length 1) and an interference radius (length $R \geq 1$). In this example, node v is not able to receive a transmission from node u if node x concurrently transmits data to node w-even though v is not adjacent to x.

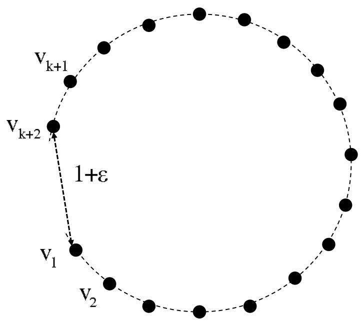

Figure 5: Example where UHI fails: nodes v_1 and v_{k+2} are separated by a path of $k + 1$ hops, but are close (distance $1+\epsilon$). Thus, concurrent transmissions of nodes v_2 and v_{k+2} may interfere at v_1 in spite of their large hop distance.

3.4 UDG with hop interference

In UHI nodes are located at arbitrary positions in $\mathbb{R}^2$. **Two nodes are adjacent if and only if their Euclidean distance is at most 1. Two nodes can communicate directly if and only if they are adjacent, and if there is no concurrent sender in the k-hop neighborhood of the receiver (in the UDG)**[2]. Clearly, this is a stark simplification since in a UDG a $(k+1)$-neighbor can be close to the receiver, see Figure 5.

3.5 General weighted graph (GWG)

GWG describes two graphs: a weighted connectivity graph G and a weighted interference graph H. For simplicity, it is often assumed that $G = H$. **A receiver v successfully receives a message from a sender u, if and only if the received signal strength (the weight of the link between u and v in G) divided by the total interference (the sum or the max of the weights of the links of concurrently transmitting nodes with a receiver v in H) is above the threshold given by the signal-to-interference-plus-noise ratio**[2]. Therefore the general weighted graph model is quite pessimistic, as it allows non-natural network topologies.

4 Algorithmic models

4.1 General problems

Besides the classic problems such as time and space complexity algorithms for sensor networks have struggle with some additional difficulties like the amount of messages to send, power saving etc[5]. Due to their special designation particular conditions have to be taken into account.

4.2 Distributed algorithms

In a distributed algorithm model every sensor node runs its own algorithm[2]. A priori, a node has only information about its own state. In order to learn more about the rest of the network, nodes repeatedly exchange messages with adjacent nodes.

4.3 Localized algorithms

A localized algorithm is a special case of a distributed algorithm. At the beginning, a node has only information about its own state. In order to learn more about the rest of the network, messages have to be exchanged[2]. In a k-localized algorithm, for some constant k, each node is allowed to communicate at most k times with its neighbors. However, a node can decide to retard its right to communicate; for example, a node can wait to send messages until all its neighbors having larger identifiers have reached a certain state of their execution. In spite of the more restricted communication model, localized algorithms can be slow. A node u might have to wait for a neighbor v to transmit all its messages, while node v in turn has to wait for its neighbor w, etc. This yields a worst-case execution time of $\Theta(n)$, where n is the number of nodes.

4.4 Local algorithms

At the beginning, each node only knows its own state. **In a k-local algorithm, for some constant k, each node can communicate at most k times with its neighbors**[2]. However, in contrast to k-localized algorithms, nodes cannot delay their decisions. In particular, all nodes process k synchronized phases, and a node's operations in phase i may only depend on the information received during phases 1 to $i-1$.

4.5 Random node distribution

The simplest and quite common way to model sensor networks is to assume a UDG in combination with a uniform node distribution in the 2-dimensional Euclidean plane[2]. However, motivated by percolation theory, also Poisson models have been proposed. Thereby, the positions of the nodes are distributed in $\mathbb{R}^2$ according to a homogeneous Poisson point process of constant density λ per unit area. While these random models may be fine to prove the performance of an algorithm, for correctness and robustness issues, a more pessimistic model should be preferred, e.g., a worst-case distribution.

5 Conclusion

As data requirements increase Sensor Networks will become commonplace. Therefore development of such systems and their working algorithms is going to be more and more important in the future. This paper has presented and compared a subjective selection of algorithmic models. Its purpose was to show that there is no optimal model that reflects all the needs best. Instead the choice of an appropriate one to the specific situation might be a preferred solution.

References

[1] T. Hänselmann: *An FDL'ed Textbook on Sensor Networks. Elektronisches Lehrbuch zur Vorlesung Sensornetze an der Universitaet Mannheim.* http://www.informatik.uni-mannheim.de/~haensel/sn_book, April 2006.

[2] S. Schmid and R. Wattenhofer: *Algorithmic Models for Sensor Networks.* 14^{th} International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS), Island of Rhodes, Greece, April 2006.

[3] D. Jeske and A. Sampath: *Signal-to-interference-plus-noise ratio estimation for wireless communication systems: Methods and analysis.* Department of Statistics, University of California, Riverside, California 92521, May 2002.

[4] F. Kuhn, T. Nieberg, T. Moscibroda, and R. Wattenhofer: *Local Approximation Schemes for Ad-hoc and Sensor Networks.* Joint Workshop on Foundations of Mobile Computing (DIALM-POMC), p. 97-103, 2005.

[5] Y. Wang, X.-Y. Li, P.-J. Wan, and O. Frieder: *Sparse Power Efficient Topology forWireless Networks. Journal of Parallel and Distributed Computing.* 35^{th} Annual Hawaii International Conference on System Sciences (HICSS'02), vol. 9, 2002.