Convergecast Algorithms for Wake-up Transceivers

Amir Bannoura[1], Leonhard Reindl[1] and Christian Schindelhauer[2]
[1]*Laboratory for Electrical Instrumentation, University of Freiburg, Georges-Köhler-Allee 106, 79110 Freiburg, Germany*
[2]*Department of Computer Networks and Telematics, University of Freiburg, Georges-Köhler-Allee 51, 79110 Freiburg, Germany*

Keywords: Wake-up Receivers, Duty-cycling Backbone, Convergecast, Online-competitive Algorithms.

Abstract: New transceiver and receiver hardware technology allow the usage of special wake-up signals, which are able to awake neighbored sensor nodes from the sleep. However, such messages need more energy e_w than those standard message transmissions e_m, when nodes are awake. Furthermore, the distance range r_w is also smaller than the distance range r_m of standard messages. Therefore, it does not completely replace duty-cycling for the convergecast problem in wireless sensor networks. We present a theoretical and practical discussion of energy-efficient algorithms for the convergecast problem. First, we present a model based on the current technology and show that without constraints on the delivery times wake-up signals are obsolete, when arbitrary long sleeping times are allowed. The wake-up graph G_w and the message graph G_m are modeled by planar r_w- and r_m-disk-graphs. Then, we give a competitive analysis for the general case, where we discuss an online Δ-convergecast algorithms bounded by competitive energy ratios. Finally, we present simulation results for these algorithmic ideas in the plane by considering the energy efficiency and the latency of data delivery.

1 INTRODUCTION

Energy is one of the most fundamental characteristics to ensure the continuous operation of Wireless Sensor Networks (WSNs). Most wireless nodes use duty cycling to conserve energy through switching off and on their wireless transceiver. The concept of wake-up receivers (Gu and Stankovic, 2004) introduces a new approach that allows the wireless transceiver to be switched off for unlimited amount of time in order to decrease the power consumption to the minimum.

The nodes switch their transceivers on when a wake-up signal is received. Transmission of data packets occurs upon activating the wireless transceivers. Despite that the wake up technology provides a solution for the energy consumption problem, transmitting a wake-up signal that is required to wake up other nodes is energy expensive compared to the data packets. As well, the wake-up distances of these signals are limited compared to the normal data communication distances. In this case, multi-hop wake-up signals are required to cover the area of a single hop of normal data communication. Since waking up the nodes periodically is considered energy expensive, a proper solution would be to maintain an active path for a while to function as a backbone for the nodes to deliver messages to the sink.

Previously, we presented several algorithms to wake-up the network from a single source node (Bannoura et al., 2015). It focuses on how to cover the network without considering any information about the nodes and their positions. However, waking up the nodes and construct a routing path from scratch each time to deliver data to the sink is not a practical approach, since this process consumes a lot of energy each time a node has data to transmit. Instead, combing the wake-up approach with duty cycling for dense networks reduce the need to transmit several wake-up signals. Based on the available information about the network, some active nodes perform duty cycle to build virtual backbones. These backbones are activated for a limited time to reduce the need for wake-up signals. In case a node has no direct communication with an active node in the backbone, wake-up signals are used to wake-up nodes until one node has a direct communication with the backbone or the sink.

Latency and energy are the two major challenges for data gathering at the sink. Despite introducing the wake-up technology to reduce energy, the end-to-end delay for packet delivery increases significantly due to the time required to wake-up the nodes in the path toward the sink. Thus, we study the problem of time

Bannoura, A., Reindl, L. and Schindelhauer, C.
Convergecast Algorithms for Wake-up Transceivers.
DOI: 10.5220/0005795601370143
In *Proceedings of the 5th International Confereence on Sensor Networks (SENSORNETS 2016)*, pages 137-143
ISBN: 978-989-758-169-4

and energy for data gathering using wake-up receivers at the sink.

2 RELATED WORK

Wake-up receivers, like the ones developed by Gamm et al. (Gamm et al., 2010), give us an alternative to the concept of duty cycling for communication in sensor networks.

An optimal solution for data aggregation is to construct a minimum connected dominating set (MCDS) of nodes to get the data to a sink. But creating such a MCDS for sensor networks is NP-hard in general graphs as well as for unit disc graphs (Lichtenstein, 1982; Clark et al., 1991). Although a polynomial-time approximation scheme (PTAS) is shown for the unit-disc version of the problem in (Cheng et al., 2003).

The wake-up receivers form an online version of the MCDS problem. In our previous work (Bannoura et al., 2015), we tried to create a technique for waking up all nodes with no knowledge about the network using a push-based epidemic rumor spreading algorithm. However, a broadcast wake-up is considered an energy expensive approach in order to construct a path between a source and a destination. In a dense network, it is better to combine the wake-up approach with the traditional duty cycling.

The main task of sensor networks is to collect information about their environment. In general, the information is gathered at the sink in a communication pattern known as convergecast. The convergecast problem focus on minimizing the required time for message delivery through minimizing the schedule time. However, Choi et al. (Choi et al., 2009) proved that finding the minimum schedule time is NP-hard for general graphs. They propose an optimal scheduling of 3(n - 2) for a minimum scheduling for a line or tree topology, where n is the number of nodes. Gandham et al. (Gandham et al., 2006) propose a distributed convergecast scheduling algorithm that requires at most 3N time slots. Through extensive simulation they showed that the actual time slot required is 1.5N. Similar result of 3N for routing in line topology is achieved by Zhang et al. (Zhang et al., 2015). For a tree routing graphs, the lower bound on the number of time slots required to complete convergecast is max { $3n_1 + \Delta$, N + 2 }, where they assume the nodes $n_1 > n_2 > .. > n_m$. $\Delta = 1$ if $n_1 = n_2$, otherwise $\Delta = 0$. Lu et al. (Lu et al., 2005) studied how to minimize the end-to-end delay communication. An approximation algorithm is proposed that achieves a bound of $d + O(k)$ for a tree and a grid topology. In arbitrary graphs, a bound of $O((d+k)\log n)$ is achieved, where d is the distance between two nodes and n is the total number of nodes.

The purpose of minimizing scheduling is to reduce the energy required for data delivery to the sink. Adjusting the communication ranges of the nodes by controlling transmission power can reduce the consumed energy. Kesselman and Kowalski (Kesselman and Kowalski, 2005) suggested a random distributed algorithm with a trade-off between latency and energy. Their approach has a latency bound of $O(\log n)$ and a minimum energy consumption of $O(n \log n)$, where n is the total number of nodes and each node can adapt to different communication ranges. Similar approach in (Yu et al., 2004) is to reduce energy consumption based on latency constrains. They considered an on-line and off-line variant for data aggregation with different communication ranges. Also, Sheng et al. (Shang et al., 2010) propose an approximation greedy algorithm for minimal convergecast and it is bounded by a constant performance ratio.

3 THE PROBLEM

The convergecast main challenges are the limited energy supply and latency for delivering packets to the sink. We consider a set of sensor nodes V is distributed in the plane. In our round model, a node is either asleep or awake. At the end of each round, a node can decide to go to sleep and set a counter to wake up again after specific round. This process is called *duty-cycling*. Another way to leave the sleep state, is to receive a wake-up message.

In our model the sensor nodes are not moving and the information about the neighbor nodes is known. The model considers two types of transmissions: First, every node can send a *data message*, which is a unicast message that contains aggregated sensor data to be forwarded to the sink. Second, each node can generate a *wake-up signal*, which can wake up a specific neighbored node. Depending on the technology, the energy and distances of these transmissions differ. Therefore, the energy of transmitting and being awake during a specific duration for a round is donated by $e_m > 0$, which also allows to transmit or receive data messages. We assume a disk-communication model, such that a node can be reach an awake neighbor in distance $r_m > 0$ with a data message. Furthermore, in a round the sensor data is small compared to the transmission overhead. So, we assume that in a round only one message is necessary to transmit all collected data, i.e. only the energy sum of a tree is necessary to send all messages to the sink.

Then, we have the wake-up signals, which wake up specific sensor nodes by carrying their addresses. Because of a special modulation they have higher energy cost $e_w > e_m$, since all available technologies need considerably higher cost, e.g. the range e_w/e_m have a value of more than 50 times. Furthermore, the transmission range r_w of a wake-up call is considered smaller, i.e. $r_w < r_m$. Current technology delimit the ratio r_m/r_w between 5 to 20 times depending on the transmission power, antenna and wake-up circuit sensitivity.

In each round, it is possible for a sensor node to sense new data. This can trigger the sensor node to send this data in this round. Also, due to the limited storage space of the node, we don't consider that the nodes are capable of storing data. So, an awake node, which has data to send, check if neighbor nodes are awake at the beginning of the round by broadcasting a status messages. Depending on this information, the node can send data messages to a subset of the awake sensors, or if no node is awake the awake node transmits wake-up messages to wake-up neighbor nodes similar to Fig 1. In the next round, the awake nodes are not necessarily receiving data and may be put back to sleep immediately. The main goal may be to wake up some connected sensors which allows the transmission of all data intended to reach the destination in this round.

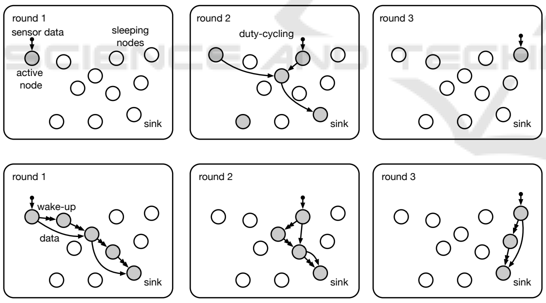

Figure 1: Duty cycling and wake-up signals.

4 THE SLEEPING BEAUTY PARADOX

We assume that the maximum distance between two neighbored nodes is bounded by r_m, since otherwise the communication network is disconnected. Now, if two neighbored nodes have a distance of more than r_w, then it is necessary that two nodes on both sides of this gap perform duty cycling.

Since our work focuses on reducing energy through waking up and performing the duty cycle, we define the average energy required per round as follows.

Definition 1. *Given a communication scheme for T rounds and let W_i be the wake-up messages and M_i be the data messages in round i. Then, the average total energy per round is defined as*

$$E_{avg} = \frac{1}{T}\sum_{i=1}^{T} |M_i|e_m + |W_i|e_w \ .$$

Now the sleepy beauty paradox is that for growing T and any sensor data the average total energy converges to 0 even without the use of wake-up messages. For this assume that all nodes wake up in rounds $f(1) < f(2) < f(3) < \ldots$ and buffer the sensor data in all other rounds. Then, the energy cost of each round is bounded by $(n-1)e_m$, since we assume perfect data aggregation. Now, if $f(i+1) - f(i)$ grows strictly monotone, then the average converges to 0 for growing T.

Proposition 1 (Sleeping Beauty)**.** *A delay tolerant sensor network with perfect data aggregation allows a duty-cycling communication scheme where the average total energy converges towards* 0 *for growing number of rounds T.*

Proof. Consider waking up all nodes at times $f(i)$ for growing function f, e.g. $f(i) = i^2$. Then the energy over all rounds is at most $n\sqrt{T}e_m$, which results in an average energy of $\frac{1}{\sqrt{T}}ne_m$, which converges towards 0. □

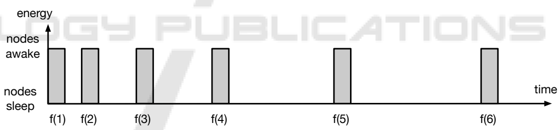

Figure 2: Increased sleep cycles make wake-up signals obsolete.

So, all nodes sleep for longer and longer times and the network becomes less and less reactive. Basically, it slows down and becomes less reactive, which is not a desirable solution for a sensor network.

In order to deal with this problem, we must set a delivery bound for all data to some number of Δ rounds. Such algorithms are called Δ-convergecast algorithms. One can see this bound as a real-time constraint on sensor data. Another motivation is that the clocks are drifting and Δ is an upper bound for synchronizing the nodes.

5 COMPETITIVE ANALYSIS

If one tries to perform a competitive analysis of this problem one faces the following problem. For an offline algorithm with full knowledge of the sensor data, the duty cycles can be set to the perfect timing that a communication networks just appears at the right time at the perfect place. In order to facilitate the delivery of data at rounds $0, \Delta, 2\Delta, \ldots$. No online algorithm can provide a reasonable bound compared to this clair-voyant solution using only duty-cycling.

Theorem 1. *Sensor networks without wake-up signals do not allow online Δ-convergecast-algorithms with bounded competitive energy ratios.*

Proof. Consider a network with $2\lceil r_m/r_w \rceil$ nodes with distance r_w on a line with the sink on one side, see Fig. 3. Now, for T rounds only one sensor measurements arrives at the other side round r. In order to meet the delay bound Δ at least one of the middle nodes has to wake up at least T/Δ times to forward the single measurement with total energy $Te_m/\Delta + 2e_m$.

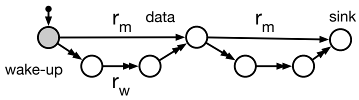

Figure 3: Wake-up signals are necessary if few sensor data arrive.

For the offline algorithm a middle node wakes up in round r which corresponds to energy $2e_m$. An algorithm with wake-up signals could have woken up the middle nodes with energy $2\lceil r_m/r_w \rceil e_w$. □

Wake-up signals allow some solution. For this, we assume from now on that there is always a wake-up path from every node to the sink, i.e. for all $u \in V$ there exists a path $(u = v_0, v_1, \ldots, v_k = s)$ such that $|v_i, v_{i+1}| \leq r_w$.

Theorem 2. *Using wake-up signals there is an online algorithm which achieves an online competitive bound of at most $4n\frac{e_w}{e_m}$. This bound is tight up to a constant factor.*

Proof. If sensor data occurs at a node it will be stored to the rounds $0, \Delta, 2\Delta, \ldots$. Then, it wakes up and wakes up all nodes on the shortest path to the sink. Then, the data is sent along the resulting shortest path tree. The overall energy cost is therefore $\frac{T'}{\Delta} n(e_w + e_m)$, if T' denotes the sum of all time intervals, where sensor data has occurred in an interval of length Δ.

The minimum offline energy needed to send data along one hop. It is at least $\frac{T'}{2\Delta} e_m$, since sensor data of two consecutive intervals could have been combined. $e_w > e_m$ implies the upper bound.

For the lower bound, consider a network, where every wake-up path uses all nodes of the network and the communication network is only one hop, see Fig. 4. As soon as duty cycling is used, no data occurs.

So, duty-cycling cannot be used and an overall cost of $\frac{T'}{\Delta} n e_w$ is necessary. □

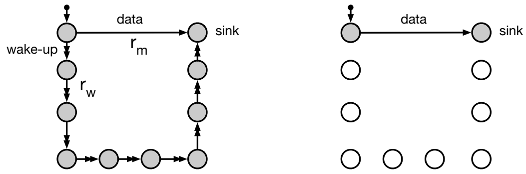

Figure 4: A worst case situation for a wake-up strategies.

Clearly such bounds are not fair, and therefore we consider a plane with densely placed sensor nodes and a fairer comparative ratio for online ratio, without a clairvoyant offline strategy. Also, these bounds are valid for sensor nodes in general positions, e.g. where the graph of possible wake-up calls G_w is much sparser than the graph of possible communication messages G_m.

6 ALGORITHMS

The theoretical analysis of the wake-up algorithms combining duty cycling needs to be throughly studied. Thus, we'll focus on the practical implementation of the algorithms. The following basic strategies for convergecast are optimal if a very high or a very small number of sensor messages needs to be handled.

Greedy Shortest Wake-up Path. Here, all sensor nodes use only wake-up calls to communicate with the sink. For this, they send wake-up calls on the shortest path towards the sink, a subset of these nodes take care of the sensor data and then all nodes go back to sleep. The sensor nodes don't consider performing duty-cycling. Thus, the energy is dominated by the number of messages s to be sent on the average and the wake-up diameter D_w of the graph, i.e. $O(sD_we_w)$.

Duty-cycling Covering Backbone is the standard approach without wake-up signals. It uses a backbone tree T_b of nodes, which is a tree in G_m, where for every node $u \in V$ there exists a node in $V_b = V(T_b)$, which can be reached within one hop. This set of node sleeps $\Delta - 1$ rounds and synchronously wakes up. Since, the nodes cover the full graph, all sensor data can be sent to the sink within Δ rounds.

Computing such a backbone is not an easy task, and it has been discussed here (Ghosh and Das, 2008;

Cardei and Wu, 2006). The average energy consumption is clearly at most $|V_b|\frac{T}{\Delta}e_m + se_m$ where s denotes the average number sensor data occurring at the sensor nodes.

Hierarchical Cut Decomposition uses the concept proposed in (Fakcharoenphol et al., 2004) to create a tree structure, where the root of the tree is the sink. The algorithm uses different radius r-cut decompositions to create several clusters. In addition, between any two nodes the algorithm achieves a stretch factor of $O(log(n))$.

We can apply this concept to create a virtual backbone depending on the different ranges between the nodes, which were created using the decomposition r-cut algorithm. The most important two r-cuts are r_w and r_m in Fig. 5. Therefore, a virtual backbone can be built from the node u_1 to any node within the range of r_m. However, choosing the nodes that are located in the outer coverage region $u_2, \ldots, u_5$ will increase the coverage of the network and reduces the required number of nodes participating in the duty-cycling backbone. When the backbone network is not active, node u_1 uses the wake-up signals to wake-up the nodes in the r_w region, then it continues until a node participating in the backbone is reached.

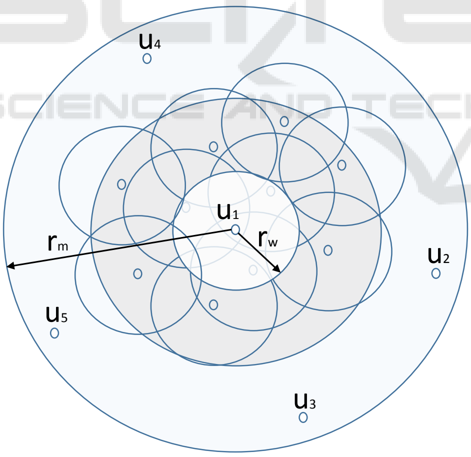

Figure 5: A hierarchical cut decomposition algorithm to create a duty-cycling backbone.

7 SIMULATIONS

We have simulated the greedy shortest wake-up path (SP) and the embedding tree (ET) algorithm to deliver messages from a source to a sink. The embedding tree algorithm uses the hierarchical cut decomposition to create a duty-cycling backbone. Although, the backbone communication range can be chosen to be any $r-cut < r_m$. Our aim is to maximize the coverage and reduce the number of participating nodes in the duty-cycling backbone. The nodes are randomly deployed in the network and the following parameters are used to implement the algorithms and measure their performances. These parameters are based on real world measurements.

Table 1: Simulation parameters.

Symbol	Description	Value
n	Number of deployed nodes	2000
l	Square area edge length	1000 m
r_w	Wake-up signal range	40 m
r_m	Data messages range	200 m
e_w	Wake-up signal energy	456 mW
e_m	Wake-up signal energy	51 mW

In Fig. 6, the wake-up algorithm constructs a path using the shortest path from each source to the sink. The nodes go back directly to sleep after waking a node located nearer to the sink location based on wake-up hops.

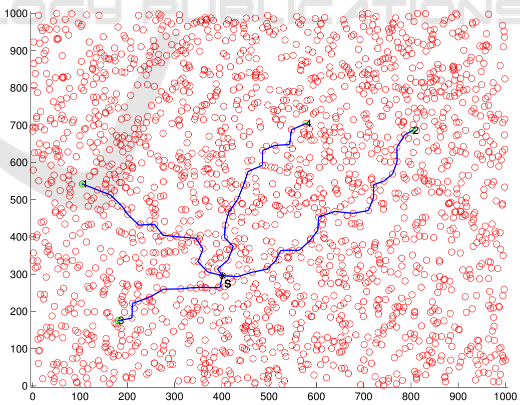

Figure 6: Shortest wake-up Algorithm.

In Fig. 7, a hierarchical cut decomposition partitions the network into clusters where the green nodes are located in the data message transmission range of other nodes. A virtual backbone is created to connect these nodes to perform a duty cycling when they are active. Nodes participating in the backbones don't go directly to sleep after transmitting data messages. In case a node sense new data, the node transmits wake-up signal to wake-up near by node or if it is located in the range of a nodes that is performing duty-cycling data messages are transmitted without the need to wake-up a chain of nodes to reach the backbone. Therefore, we can see in Fig. 7 that the source nodes transmit wake-up signals depicted in the blue continuous line until they are in the range of the backbone, then data messages are transmitted which are depicted in a black dashed line.

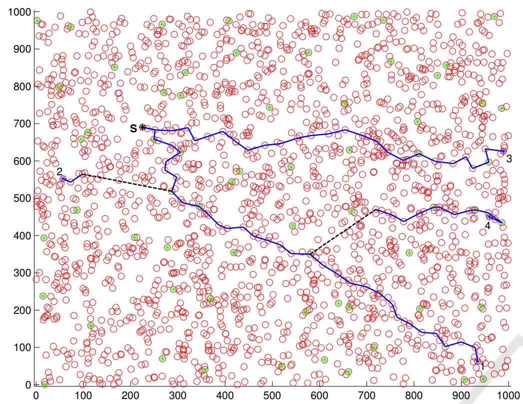

Figure 7: Embedding Tree Algorithm.

We measured the performance of the algorithms according to message delivery delay and power consumption. Fig 8 shows the delay needed to transmit a message from a source to the sink. In an inactive and slow networks where events happen infrequently, the delay is limited to waking the nodes on the shortest path to the sink. When the density of events increases, message delivery will occur on the backbone which reduces the need to transmit wake-up messages. The message delay is measured based on the number of hops to reach the sink. Thus, the delay in the shortest wake-up algorithm depends on the diameter of the network, whereas the delay in the embedding tree depends on the number of active nodes. The delay achieved for the embedding tree (duty 50%) has a better performance than the shortest wake-up and other embedding tree with different duty-cycles since the nodes stay longer period active and cover higher transmission ranges.

For power consumption in Fig 9, we randomly generated 10 events in the network in different rounds to measure the power needed to deliver messages to the sink. Power consumption using the shortest path grows linearly according to the number of messages and the wake-up diameter described in section 6. The shortest wake-up algorithm performs efficiently compared to the embedding tree algorithm due to the fact that the nodes go directly to sleep and don't waste time in idle listening in the network. However, upon increasing the events and the network become more

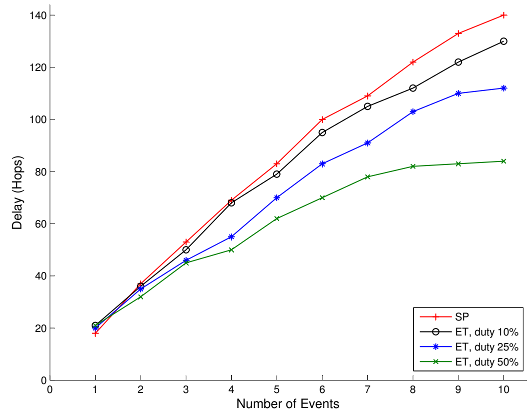

Figure 8: Message delivery delay.

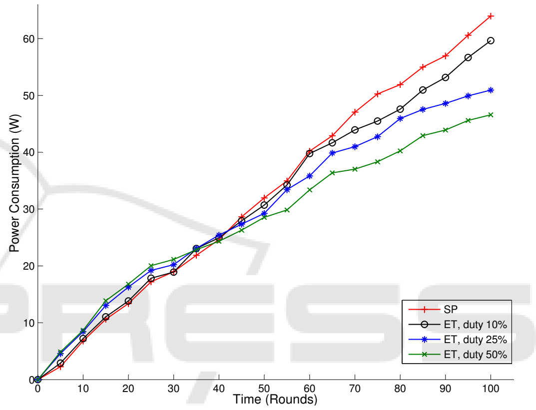

Figure 9: Network power consumption.

active, the nodes will use the backbone network to deliver data messages which reduces the need to transmit wake-up signals. Since idle listing in an active network consumes less energy than waking-up the path to the sink. Therefore, the shortest wake-up algorithm can't compete with embedding tree algorithm on the long run for a dense active network.

8 CONCLUSIONS AND FUTURE WORK

The development of wake-up receivers is an alternative approach for sensor networks. Using wake-up receivers decreases energy consumption to the minimum, where the transceiver is activated only on demand by receiving wake-up signals or through sensing new data.

In this work, we study the problem of convergecast for wake-up transceivers, where all the nodes try to deliver messages to the sink. The challenge is to decrease the latency for delivering messages because using the wake-up transceivers will keep the network for longer periods in sleep, which makes the network less reactive. Thus, the convergecast algorithms have to set a delivery bound of Δ rounds to deliver messages to the sink in order to maintain an active network.

We proposed a greedy shortest wake-up path and embedding tree duty cycle covering backbone. The performance for combing both the wake-up and duty-cycle for a dense network lowers the energy and reduces the delay compared to the greedy wake-up algorithm.

Furthermore, the behavior of the wake-up transceiver increases the time required to wake-up the nodes, which increases the latency of message delivery. A theoretical analysis for combining duty-cycling with the wake-up transceiver has to be extensively studied. Since, the competitive ratio of the algorithms has to be compared with the competitive ratio of the offline algorithms.

REFERENCES

Bannoura, A., Ortolf, C., Reindl, L., and Schindelhauer, C. (2015). The wake up dominating set problem. *Theoretical Computer Science*, 8243(0):35 – 50.

Cardei, M. and Wu, J. (2006). Energy-efficient coverage problems in wireless ad-hoc sensor networks. *Computer Communications*, 29(4):413 – 420. Current areas of interest in wireless sensor networks designs.

Cheng, X., Huang, X., Li, D., Wu, W., and Du, D.-Z. (2003). A polynomial-time approximation scheme for the minimum-connected dominating set in ad hoc wireless networks. *Networks*, 42(4):202–208.

Choi, H., Wang, J., and Hughes, E. A. (2009). Scheduling for information gathering on sensor network. *Wirel. Netw.*, 15(1):127–140.

Clark, B. N., Colbourn, C. J., and Johnson, D. S. (1991). Unit disk graphs. *Discrete Math.*, 86(1-3):165–177.

Fakcharoenphol, J., Rao, S., and Talwar, K. (2004). A tight bound on approximating arbitrary metrics by tree metrics. *Journal of Computer and System Sciences*, 69(3):485 – 497. Special Issue on {STOC} 2003.

Gamm, G., Sippel, M., Kostic, M., and Reindl, L. (2010). Low power wake-up receiver for wireless sensor nodes. In *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2010 Sixth International Conference on*, pages 121–126.

Gandham, S., Zhang, Y., and Huang, Q. (2006). Distributed minimal time convergecast scheduling in wireless sensor networks. In *Distributed Computing Systems, 2006. ICDCS 2006. 26th IEEE International Conference on*, pages 50–50.

Ghosh, A. and Das, S. K. (2008). Coverage and connectivity issues in wireless sensor networks: A survey. *Pervasive and Mobile Computing*, 4(3):303 – 334.

Gu, L. and Stankovic, J. (2004). Radio-triggered wake-up capability for sensor networks. In *Real-Time and Embedded Technology and Applications Symposium, 2004. Proceedings. RTAS 2004. 10th IEEE*, pages 27–36.

Kesselman, A. and Kowalski, D. (2005). Fast distributed algorithm for convergecast in ad hoc geometric radio networks. In *Wireless On-demand Network Systems and Services, 2005. WONS 2005. Second Annual Conference on*, pages 119–124.

Lichtenstein, D. (1982). Planar formulae and their uses. *SIAM Journal on Computing*, 11(2):329–343.

Lu, G., Sadagopan, N., Krishnamachari, B., and Goel, A. (2005). Delay efficient sleep scheduling in wireless sensor networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 4, pages 2470–2481 vol. 4.

Shang, W., Wan, P., and Hu, X. (2010). Approximation algorithm for minimal convergecast time problem in wireless sensor networks. *Wireless Networks*, 16(5):1345–1353.

Yu, Y., Krishnamachari, B., and Prasanna, V. (2004). Energy-latency tradeoffs for data gathering in wireless sensor networks. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1, page 255.

Zhang, H., Osterlind, F., Soldati, P., Voigt, T., and Johansson, M. (2015). Time-optimal convergecast with separated packet copying: Scheduling policies and performance. *Vehicular Technology, IEEE Transactions on*, 64(2):793–803.