Convex Groups for Self-organizing Multi-sink Wireless Sensor Networks

Olga Saukh
Structural Engineering Research Laboratory, EMPA, Switzerland
olga.saukh@empa.ch

Robert Sauter
Universität Bonn, Bonn and Fraunhofer IAIS, St. Augustin, Germany
sauter@cs.uni-bonn.de

Pedro José Marrón
Universität Bonn, Bonn and Fraunhofer IAIS, St. Augustin, Germany
pjmarron@cs.uni-bonn.de

***Abstract*—Mobile ad-hoc networks have long been proposed for rescue scenarios to support and coordinate the efforts of helpers. Low power wireless sensor networks are a natural extension of this approach. They can provide valuable environmental data enriched with location information to deepen the insight of the operational area. Gateways between these two communication paradigms are necessary to facilitate the collaboration of both systems. We study such multi-sink wireless sensor network scenarios and explore the use of *convex groups* to efficiently disseminate location dependent information, for example, for query distribution. Convex groups show significant advantages in terms of message overhead compared to straightforward approaches without sacrificing connectivity. They do not exhibit high computational complexity and handle node mobility as well as gateway mobility gracefully.**

I. Introduction

Most newly developed applications for wireless sensor networks require the cooperation of a wireless sensor network (WSN) with a more powerful mobile ad-hoc network (MANET). Such applications rely on the use of contextual data extracted from the sensor network. Either each MANET device communicates directly with its surrounding sensor network, which requires an additional WSN-compatible communication adapter, or some devices act as gateways (sinks) and supply the others with context information over the ad-hoc network.

The combination of these two different types of systems – large scale and extremely resource-constrained sensor network on the one hand and a powerful ad-hoc network on the other hand – creates new challenges that need to be addressed from the ground up. The most fundamental challenge deals with the development of efficient communication and cooperation paradigms between these network types.

In the AWARE project[1], several applications that involve the cooperation between MANETs and WSNs are studied. One example is the fire fighting scenario where, in case of fire, sensor nodes are deployed with unmanned aerial vehicles (UAVs) to provide context information. The collected environmental data is used to assist the operation of fire fighters and rescue units as well as the mission planning of the available UAVs.

This scenario requires a flexible architecture which enables easy access to context data and allows to incorporate various types of MANET devices used in this setting. Preinstalled cameras, UAVs, fire trucks and fire fighters equipped with PDAs or laptops have to exchange large amounts of data over the ad-hoc network. Some of these devices also serve as gateways to the WSN and provide context data to the other MANET devices. This results in a number of challenges such as multiple potentially highly mobile sinks where the speeds vary from that of a typical pedestrial to that of a flying UAV. Moreover, the deployed sensor network might be disconnected due to holes in the deployment, sink mobility or destruction of sensor nodes by fire.

The data of the WSN has to be annotated with location information to be of use in this scenario. Thus, we assume that sensor nodes know their positions, which can be assigned by a UAV prior to the deployment. This makes it possible to query a part of the WSN by specifying an area of interest.

In this paper, we concentrate on providing efficient access of MANET devices to context information provided by a WSN with the help of gateway nodes. The reduction of the message overhead and the overall energy consumption of the WSN is the foremost goal as well as the support for gateway and sensor node mobility. The following challenges are considered: 1) scoping of query dissemination to areas of interest; 2) efficient network reconfiguration due to topology changes; 3) efficient handling of node and especially sink mobility.

We solve these challenges by defining convex groups that encapsulate the coverage information of the WSN. The convex group of each subtree is calculated on-the-fly along the routing tree to the nearest sink. Moreover, changes in the routing tree, e.g., due to node mobility, require recalculation only of a limited number of groups and, therefore, impact only a limited part of the network. Additionally, it is possible to limit the space complexity of convex groups to a constant value by lossy compression, which can reduce the message overhead but does not impair the correctness of query dissemination.

We discuss several alternative approaches and argue that convex groups are a low-cost and practical abstraction which enables efficient cooperation of multiple sinks. We propose a new approach that leverages position information to optimize cooperation among gateways in multi-sink scenarios by hierarchical spatial scoping. This enables efficient query dissemination and supports node and sink mobility taking position inaccuracy, topology changes and poor link quality into account.

[1]The AWARE project (IST-2006-33579, http://www.aware-project.net/) is funded by the European Union

The rest of this paper is structured as follows. In Section II we give an overview of related work. Section III details the motivating scenario and the resulting challenges addressed in this paper. The definition of convex groups and the required algorithms are presented in Section IV. We evaluate our approach under a variety of settings in Section V, and Section VI concludes the paper and provides an insight on our next steps.

II. Related Work

There are a number of related approaches in the area of efficient partitioning of a sensor network among multiple sinks and optimization techniques for data acquisition.

Multi-sink partitioning: Many real-world applications face the problem of multi-sink routing in sensor networks. However, comparatively little research has been conducted in this area [1]. The simplest possible solution is for each sink to disseminate messages to the whole network. However, this approach is redundant and the per node overhead increases linearly with the number of sinks [2]. The algorithm presented in [2], [3] describes Voronoi scoping – a distributed algorithm to constrain the dissemination of messages from different sinks by introducing Voronoi clusters. Every node belongs to the Voronoi cluster of the closest sink, where "closest" depends on the underlying distance metric. In [1] the authors generalize Voronoi scoping by introducing the so-called Logical Graph Model, where multiple sinks are seen as a single logical sink. The constructed graph allows for easy adaptation of algorithms that have been designed for single sinks to multi-sink data acquisition. Other related scoping techniques include TTL scoping and geographic scoping that consider partitioning of sensor network based on the time-to-live for a message and the Euclidean distance to the nearest sink respectively.

The approach presented in this paper can use any of the discussed scoping metrics or any other tree-forming routing metric to partition the network. Additionally, our approach leverages the knowledge of node positions to enable efficient querying of subregions of the WSN. Therefore, the hierarchical convex groups allow for extended scoping that is necessary to route the data between the closest sink and the subregion of interest.

Query dissemination: Querying is typically done through techniques such as flooding [4], minimum broadcast tree algorithms [5], or probabilistic algorithms such as gossiping [6]. The approach presented in this paper differs from these approaches by using the knowledge of the sensor node coordinates to scope query dissemination messages with convex groups. Moreover, convex groups allow to limit the flooding between the sink and the destination subregion and, therefore, to prolong network lifetime.

Since we assume that all nodes know their coordinates, we also have to consider geographic routing approaches like GPSR [7] in the context of the gateway cooperation problem. However, since these approaches do not tackle specifically the problem of tree-based collection and dissemination, they do not provide an efficient solution. Additionally, the protocols might either fail on random network topologies or are very complex and require the computation of a planar subgraph of the underlying connectivity graph. Moreover, geographic routing might fail if coordinate information is inaccurate. The convex groups approach overcomes these limitations, while still providing support for sensor node and sink mobility, and the algorithm is simpler compared to geographic routing.

III. Gateway Cooperation

In the following subsections we describe the scenario that motivates the need for gateway cooperation when sensor nodes know their coordinates and describe the advantages of the concept of convex groups.

A. Fire Fighting Scenario

AWARE is a three-year European project that has been established to study the potential for self-organising and collaborative sensor networks. In particular, the cooperation of a wireless ground sensor network comprising static as well as mobile devices with unmanned aerial vehicles (UAVs). In this scenario it is essential to establish an efficient communication and cooperation platform that is able to self-organize, adapt to changes and provide support in case of accidents.

The primary application of the AWARE platform is a fire fighting scenario in which the UAVs must trigger an alarm if a fire is detected and start to deploy the sensor nodes, which measure temperature, gas level and other environmental data that is transmitted to the fire fighters and mission coordinators.

The described scenario involves two rather different communication standards: UAVs and PDAs communicate over IEEE 802.11g in ad-hoc mode and sensor networks usually use low power and cheap transceivers for example employing the IEEE 802.15.4 standard. Moreover, the devices in the high bandwidth parts and in the low bandwidth parts of the AWARE network use different communication paradigms: address-centric and data-centric respectively. Therefore, there is a strong need for gateway devices equipped with both communication interfaces to cooperate efficiently in order to achieve the goals of AWARE.

B. Gateway Cooperation

The first problem to solve is to define the areas every mobile gateway is responsible for in order to avoid that all queries have to be forwarded to all sensor networks via all gateways. There are two groups of approaches to define such areas: *top-down* and *bottom-up*.

Top-down approaches assign the network partitioning task to sink nodes. As soon as sink nodes define their areas of responsibility, the sensor nodes are notified by every sink providing a description of the area. Every sensor node is then able to select the area it belongs to. For example, the sink nodes might partition the target area by Voronoi tessellation to define their areas of responsibility. The main disadvantage of this group of approaches is that the partitioning does not take into account possible absence of connectivity in a sensor network within each partition. This happens due to non-uniform deployment of nodes in certain areas, routing holes due to environment characteristics, unstable and asymmetric communication links between the nodes and node mobility. Therefore, top-down area partitioning often causes unreachable nodes in the network.

The other group of approaches for the definition of the areas of responsibility for every sink node is based on the cooperation of sensor nodes within the network. These are *bottom-up* approaches. Based on the own position each sensor node selects the nearest gateway and sends a notification specifying its position. This group of approaches is very attractive because it is possible to preserve the reachability between a sensor node and the sink it belongs to. The approach proposed in this paper belongs to this group of approaches.

In the next section we give the definition of convex groups and describe the supporting algorithms.

IV. Distributed Convex Groups

A. *Establishing Convex Groups*

Convex groups approximate the areas of responsibility for every sink. This allows more efficient querying of a part of a sensor network while avoiding unnecessary flooding of the complete network with query messages.

A sensor network is usually modelled as an undirected graph $G = (V, E)$ with an edge between any two nodes that can communicate with each other. We propose the idea of constructing hierarchical convex groups along the routing tree in order to abstract the target area. Consider a set of s nodes $V_{[s]} \subseteq V$. We say that a polygon $P_{[n]}$ defines a *convex group* over $V_{[s]}$ if it is a convex polygon of n vertices, that covers all s nodes in $V_{[s]}$. If there is no limit on the number of vertices the polygon $P_{[n]}$ may comprise, then the minimum $P_{[n]}$ coincides with the convex hull $C_{[m]}$ built over the set of nodes $V_{[s]}$ ($n = m$). Moreover, it is possible to define a *compressed* polygon $P_{[n']}$ which contains the convex hull but comprises less vertices ($n' < m$) and which has, therefore, a larger area. This polygon might consist of vertices with coordinates different from any of the actual sensor nodes. Compression may be used to limit the amount of storage needed for polygon descriptions. In the evaluation section we will show that the constructed convex groups are a good approximation of the covered area.

The operator $g : V_{[s]} \to P_{[n]}$ constructs a convex group $P_{[n]}$ on the set of nodes $V_{[s]}$ without compression. The construction of convex groups along the routing tree corresponds to the divide and conquer approach of constructing a convex hull. However, the merge step of this approach assumes that the convex polygons to be merged are disjoint. This cannot be guaranteed when convex groups are built over a routing tree in contrast to an explicit divide operation where all vertices are sorted based on their x-coordinates and the resulting sequence is cut in halves. Therefore, we adopted the Rotating Calipers approach described in [8], [9] to solve this problem. This algorithm involves no backtracking and does not require that the polygons do not intersect. The convex hull algorithm including Rotating Calipers merge step for possibly intersecting polygons runs in $O(n \log n)$ time and requires $O(n)$ space.

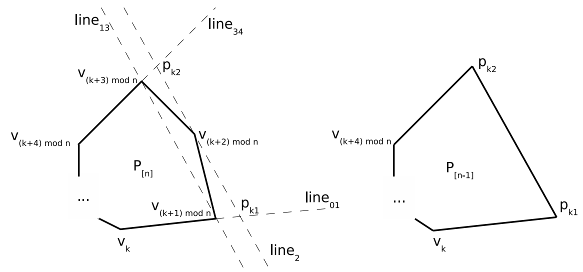


Fig. 1. Compression step: $c : P_{[n]} \to P_{[n-1]}$

For the distributed construction of convex groups along the routing tree, every parent node receives the convex group information from all its children and merges them together with the addition of the parent node itself. This procedure constructs a new composite convex group, which is forwarded further. This allows the construction of a hierarchy of convex groups. The distributed procedure requires $O(1)$ messages per sensor node, $O(dm)$ space and runs in $O(dm)$ time, where d is the maximum node degree in the network and m is the maximum number of nodes in a convex polygon.

Using the structure of the routing tree has a number of advantages. First, the constructed convex groups encapsulate connectivity information. Moreover, depending on the routing metric used every node belongs to the convex group of the best parent in the sense of link quality, number of required transmissions, etc. Second, the routing tree structure provides the possibility to define a natural hierarchy over the convex groups. In the evaluation section we will argue that this hierarchy is essential for node mobility and allows for efficient routing and data dissemination in mobile scenarios.

B. *Compression of Convex Groups*

Since the number of bytes to be transmitted with one message is very limited (usually 29 bytes in TinyOS messages) and main memory is scarce on sensor network devices, it might be necessary to limit the number of vertices that approximate a convex group accepting a small loss of quality. To achieve this, the algorithm transforms the merged polygon $P_{[n]}$ in another one $P_{[k]}$ ($k < n$) if the number of vertices in a merged polygon exceeds a predefined threshold. We refer to this step as *compression*.

Consider a convex polygon $P_{[n]}$ which is a convex hull of n vertices over the set of vertices $V_{[s]}$ on the plane. We define a compression operator $c : P_{[n]} \to P_{[n-1]}$ which converts the given convex polygon of n vertices into a convex polygon of $n-1$ vertices which contains $P_{[n]}$. The Algorithm 1 describes the compression procedure performed by the operator c (also see Fig. 1). The operator c has the following obvious properties:

- **Convexity**: $c(P_{[n]})$ is convex if $P_{[n]}$ is convex
- **Inclusion**: $P_{[n]} \subseteq c(P_{[n]})$
- **Iterative applicability**: $P_{[n]} = c(P_{[n+1]})$

These properties allow us to iteratively apply the compression operator at each step along the routing tree in order

Algorithm 1 Compress $P_{[n]}$

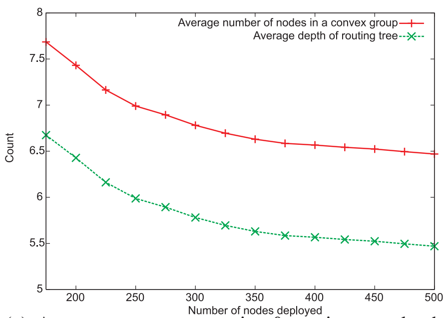
Input: convex $P_{[n]}$, $n > 3$
Output: convex $P_{[n-1]} \supseteq P_{[n]}$
  $S \leftarrow \infty, m, p_1, p_2$
  for in $P_{[n]}$: $k = 0..n-1$ **do**
    $line_{01} \leftarrow line(v_k, v_{(k+1) \bmod n})$
    $line_{34} \leftarrow line(v_{(k+3) \bmod n}, v_{(k+4) \bmod n})$
    $line_{13} \leftarrow line(v_{(k+1) \bmod n}, v_{(k+3) \bmod n})$
    $line_2 \leftarrow line(v_{(k+2) \bmod n}) \,||\, line_{13}$
    $p_{k1} \leftarrow line_{01} \cap line_2$
    $p_{k2} \leftarrow line_{34} \cap line_2$
    $S_k \leftarrow area_{\Delta}(v_{(k+1) \bmod n}, p_{k1}, v_{(k+2) \bmod n}) + area_{\Delta}(v_{(k+2) \bmod n}, p_{k2}, v_{(k+3) \bmod n})$
    if $S > S_k$ **then**
      $S \leftarrow S_k$, $p_1 \leftarrow p_{k1}$, $p_2 \leftarrow p_{k2}$, $m \leftarrow k$
    end if
  end for
  in $P_{[n]}$: $v_{(m+1) \bmod n} \leftarrow p_1$
  in $P_{[n]}$: $v_{(m+3) \bmod n} \leftarrow p_2$
  $P_{[n-1]} \leftarrow$ in $P_{[n]}$: remove $v_{(m+2) \bmod n}$

to limit the number of vertices in the convex groups. The computational complexity of one iteration step is $O(n)$ for an input polygon $P_{[n]}$.

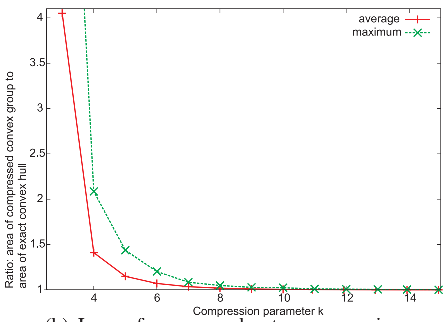
C. Properties of Convex Groups

The convex groups have the following properties:

Scoping: Every convex group describes a subregion of the monitoring area which is the scope of the convex group. When a query concerning this subregion is disseminated by the closest sink node, only the convex groups with higher hierarchy level that include this region are affected. The response to the query is propagated along the reverse path to the closest sink. Scoping also allows more efficient in-network data aggregation. The aggregation function might decide to



(a) Average convex group size & routing tree depth



(b) Loss of accuracy due to compression

Fig. 2. Convex group statistics

merge the scopes into a larger convex group or forward the data of both convex groups separately.

Scalability: The introduction of multiple sinks increases the scalability of sensor networks [2]. Additionally, scoping provides a practical method for specifying subregions of any size and due to compression provides a scalable abstraction.

Maintenance overhead: We distinguish between the case when a parent node knows its children and when it does not. In the first case, a parent waits for the data of all its children and then forwards the complete convex group to its parent. This results in minimum message overhead for the transmission of convex group information, but requires more memory on the parent and a larger number of messages for the construction and maintenance of the routing tree itself. This overhead is acceptable if this data is used by other parts of the system. In the second case, a heuristic must be used to determine the delay until the convex group is transmitted to the parent. This may result in a larger number of update messages. However, even in the worst case, the network-wide number of messages is in $O(hl)$ where h is the maximum depth of the routing tree and l is the number of leaf nodes.

The maintenance of convex groups requires every parent node to store the vertices of its current convex group. If support for node mobility is required, every parent node additionally has to store the convex groups of its children in order to recalculate its convex group without having to request this information for every update it receives.

Mobility: We distinguish sink and sensor node mobility. If a sensor node moves, position changes are propagated only as far as the convex groups are affected. Mobile sinks have a greater influence on the sensor network topology due to more significant changes in the routing tree structure. Our evaluation results show that even in this case the number of update messages is reduced by the convex group approach.

Reachability: The algorithm guarantees that a query is disseminated to all nodes in its scope. Top-down approaches, e.g., a Voronoi tesselation of the WSN between multiple sinks, cannot guarantee reachability due to lack of knowledge about the connectivity of the network.

Adaptation: The compression parameter k can be adapted based on different requirements or system characteristics:

Available memory and bandwidth: By limiting k the linear space complexity can be reduced to a constant one.

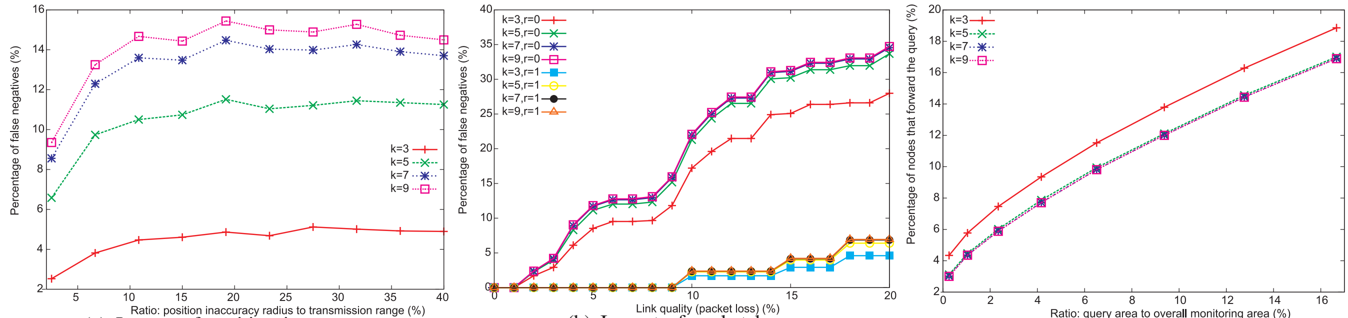
Position accuracy: Since the area of a convex group is over-estimated by compression, the number of nodes falsely considered to be outside of a convex group is reduced.

Link quality: The over-estimation of areas due to compression reduces the error incurred by packet losses.

Network density: The amount of memory required to store convex groups of children is kd (d is the network density). Lower values of k are advisable to reduce the memory overhead for high densities.

V. Evaluation

We evaluated our approach by extensive simulations varying the number of sensor nodes and the number of sinks. The



(a) Impact of position inaccuracy

(b) Impact of packet loss

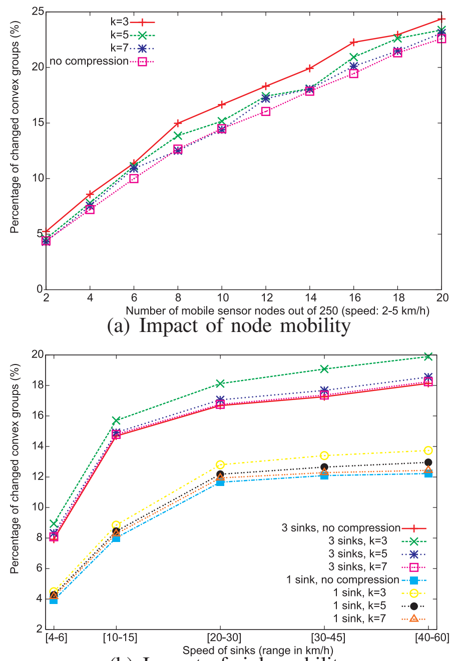
(c) Efficiency of query dissemination

Fig. 3. Impact of adverse conditions on the quality of convex grouping and efficiency of query dissemination

nodes are uniformly deployed in a rectangular region of 1200 m × 800 m. Every sensor node has a transmission range of 120 m. We used a Random Waypoint movement model [10] provided by the CANU Mobility Simulation Environment[2] to simulate sensor node and sink mobility. Input parameter settings for user mobility in rescue mission were chosen as described in [10]. The typical speeds for UAVs (40-60 km/h) are taken from the AWARE specification document and were used in the real-world AWARE experiments in April 2008 in Utrera, Spain. The mobility simulations lasted 30 simulation minutes each with update step of 10 seconds.

The number of convex groups always equals the number of deployed nodes. In Fig. 2a) we evaluated the average number of sensor nodes in a group and the average depth of the constructed routing tree. We used the Shortest Path First routing metric [11] to build a routing tree. However, the convex groups can be combined with any other tree-based routing metric, e.g., [12], [13]. The average values are calculated over 20 deployments for a fixed number of sensor nodes.

[2]http://canu.informatik.uni-stuttgart.de/mobisim/



(a) Impact of node mobility

(b) Impact of sink mobility

Fig. 4. Impact of mobility on the maintenance overhead of convex grouping

Fig. 2b) shows the average and maximum error obtained for different values of the compression parameter k. Here we varied the number of nodes in the deployment from 150 to 300 and simulated a total of 200 topologies. The compression error decreases exponentially with increasing k. Therefore, the value of k can be chosen based on user requirements on the quality of convex grouping.

Exact positions of sensor nodes can be assigned only manually. All other techniques like GPS receivers, localization algorithms [14] or assignment of node positions by a UAV result in inaccuracy of node positions. In Fig. 3a) we evaluate the influence of position inaccuracy on the quality of convex grouping. The results show, that the use of compression allows to hide inaccuracy of node positions to some extent. We used 200 deployments of 200 nodes for this plot. The positioning error is uniformly distributed within the given error radius. We evaluated the number of sensor nodes that due to position inaccuracy do not belong to the convex groups they must belong to. The positioning error of 40% (normalized by the transmission range) results in 16% of nodes being outside of their respective convex groups on average.

Packet losses are unavoidable in sensor networks and make many algorithms hardly applicable to real-world deployments. We evaluated the influence of losses of packets containing convex group information on the correctness of the convex groups. We count the number of nodes that lie outside of a convex group they should belong to. Fig. 3b) plots the percentage of such nodes versus the link packet loss rate if each packet is sent once ($r = 0$) and if one retransmission in case of transmission failure ($r = 1$) is used. If a packet is lost, the information about the whole subtree is lost. Application of one retransmission reduces the convex grouping error from 35% to 5.5% on average with link packet losses of 20%. The use of compression slightly reduces the convex grouping error as well. The average values are built over 10 topologies of 300 nodes.

The application of convex grouping allows to reduce the number of broadcasts required to disseminate query information. Queries refer to a certain subregion of the deployment area – the area of interest – a square area of a certain size in the simulation. In Fig. 3c) we show the dependency between the size of the subregion addressed by the query (relative to the overall deployment area) and the number of nodes that broadcast the query to their dependent convex groups. A low value of the compression parameter k slightly increases the number of nodes that need to broadcast the query due to the inaccuracy of the convex hull approximation resulted by compression. This evaluation result shows the average of 20 deployments of 250 sensor nodes for every point in the figure. This graph shows, that for both single-sink and multiple-sink scenarios the application of convex groups allows to reduce the number of message broadcasts for query dissemination considerably. Compared to flooding, which is usually used for query dissemination, e.g., in TinyDB [15], and requires all sensor nodes to broadcast a message, convex groups reduce the number of broadcasts considerably and, therefore, prolong the lifetime of the sensor network.

Many algorithms for sensor networks concentrate on static or low mobility applications. However, in rescue and civil security/disaster management scenarios like AWARE, it is important to take node mobility into account and to use algorithms that provide additional support for node mobility. In Fig. 4a), we plot the dependency between the number of mobile sensor nodes that represent the fire fighters. As can be seen from the graph, we experimented with 2 to 20 mobile sensor nodes out of 250 which results in up to 25% of changes in convex groups. Notice, that lower values of the compression parameter k result in a slightly higher percentage of group changes. The reason for this behaviour lies in the distributed computation of the convex hulls and the application of the compression step at each level. Each convex group is composed of two or more subgroups with the exception of leaf nodes. If a node moves within a composite convex group, it does not influence it, since it is not part of the boundary. However, if compression is applied, the movement of the node may influence the position of vertices which are on the boundary of a subgroup and of the composite group. Therefore, even nodes traveling inside of a convex group might change the boundary of this convex group.

Finally, we explored the influence of sink mobility on the algorithm. In Fig. 4b) we consider the case when 1 or 3 sinks move with different speeds. We varied the speeds of mobile sinks from the range of a typical pedestrian (4-6 km/h) to a flying UAV (40-60 km/h) and calculated the average number of convex group changes per update interval (10 sec). Lower values of k result in a slightly higher number of group changes for the same reason as explained above.

The evaluation results of scenarios involving mobile devices show the advantage of using the hierarchy of the routing tree. Only changes that affect a convex group are forwarded. Since the scope of the convex group increases along the routing tree, the probability that a change on a lower level influences a convex group decreases and thus a change does not usually propagate to the sink. The worst case message complexity of convex grouping occurs when node movement causes updates of all convex groups and is comparable to the message complexity of the described related approaches. However, convex groups perform much better in the average case.

VI. Conclusions and Future Work

In this paper we present the notion of convex groups for multi-sink wireless sensor networks – groups of nodes included in a convex region in the deployment area. Every node stores the spatial summary of the region it is responsible for. We provide algorithms that allow to efficiently build convex groups and support lossy compression to reduce the amount of spatial summary information and thereby the message complexity. This makes convex groups a powerful abstraction that allows for efficient querying of a sensor network and supports both sensor node and sink mobility, which is important for rescue missions and civil security operations.

As a part of future work, we plan to extend our 2-dimensional convex groups to the 3-dimensional case. This is important for rescue scenarios that need to operate in buildings or underground facilities, where the presence of the third dimension cannot be ignored.

References

[1] A. Das and D. Dutta, "Data acquisition in multiple-sink sensor networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 9, pp. 82–85, 2005.

[2] H. Dubois-Ferriere, D. Estrin, and T. Stathopoulos, "Efficient and practical query scoping in sensor networks," in *IEEE Inter. Conf. on Mobile Ad-hoc and Sensor Systems (MASS)*, 2004.

[3] H. Dubois-Ferriere and D. Estrin, "Efficient and practical query scoping in sensor networks," Tech. Rep., 2004.

[4] R. G. C. Intanagonwiwat and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in *Proc. of the 6th Inter. Conf. on Mobile Computing and Networking*, 2000.

[5] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Energy-efficient broadcast and multicast trees in wireless networks," *Mobile Networks and Applications*, vol. 7, no. 6, pp. 481–492, 2002.

[6] D. Kempe and J. M. Kleinberg, "Protocols and impossibility results for gossip-based communication mechanisms," in *Proc. of the 43rd Symposium on Foundations of Computer Science (FOCS)*, 2002.

[7] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, 2000, pp. 243–254.

[8] M. Shamos, "Computational geometry," PhD thesis, 1978.

[9] G. T. Toussaint, "Solving geometric problems with the Rotating Calipers," in *Proc. of IEEE MELECON*, 1983. [Online]. Available: http://cgm.cs.mcgill.ca/~godfried/publications/calipers.ps.gz

[10] I. Stepanov, J. Haehner, C. Becker, J. Tian, and K. Rothermel, "A meta-model and framework for user mobility in mobile networks," in *In Proc. of the 11th Inter. Conf. on Networking (ICON)*, 2003.

[11] D. S. J. D. Couto, D. Aguayo, B. A. Chambers, and R. Morris, "Performance of multihop wireless networks: shortest path is not enough," *SIGCOMM Computer Communications Review*, vol. 33, pp. 83–88, 2003.

[12] O. Saukh, P. J. Marrón, A. Lachenmann, M. Gauger, D. Minder, and K. Rothermel, "Generic routing metric and policies for WSNs," in *Proc. of the 3rd European Workshop on Wireless Sensor Networks*, 2006.

[13] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wirel. Netw.*, vol. 11, no. 4, pp. 419–434, 2005.

[14] K. Langendoen and N. Reijers, "Distributed localization in wireless sensor networks: a quantitative comparison," *Comput. Networks*, vol. 43, no. 4, pp. 499–518, 2003.

[15] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TinyDB: An acquisitional query processing system for sensor networks," *ACM Transactions on Database Systems*, vol. 30, pp. 122–173, 2005.