Constrained-based Multiple Sink Placement for Wireless Sensor Networks

Joakim Flathagen*‡¶, Øivind Kure‡¶ and Paal E. Engelstad§¶
*Norwegian Defence Research Establishment, ‡Q2S NTNU, ¶UNIK, §University of Oslo

***Abstract*—A wireless Sensor Network (WSN) consists of many low-cost and energy-constrained sensing nodes. One method that offers a great potential for improving both the lifetime and the durability of WSNs is to deploy multiple data sinks instead of the standard approach relying on just one sink. In this paper we focus on multiple sink deployment problems and discuss different methods to estimate the optimal placement of a given number of sinks. Most previous works study unconstrained sink node placement, assuming that the sinks can be placed anywhere. In practice, there may be areas which are occupied by obstacles, or are beyond wireless range, and therefore not viable for sink placement. Our method inherently considers deployment constraints by inspecting the routing topology and therefore avoids connection black holes when proposing optimal sink locations. We have used an anycasting tree-routing scheme, and have performed extensive simulations in a wide range of realistic scenarios. The results show that a constraint-based deployment algorithm is paramount to get the full potential of multiple sink WSNs.**

I. Introduction

A wireless Sensor Network (WSN) consists of many small and low-cost sensing nodes. The two basic challenges in WSNs are *energy efficiency*, due to the battery-powered sensors, and *scalability*, due to a potential high number of devices needing to interoperate. In this paper we aim to prolong the network lifetime and improve the scalability by deploying *multiple sinks*. In addition to reducing the average path length between a sensing node and the corresponding data sink, the use of multiple sinks also provides energy fairness by load balancing. The method also gives redundancy if one of the sink-nodes should fail due to energy shortage, or if it is vandalized or stolen.

While finding the optimal number of sinks is by nature an off-line problem mainly constrained by deployment cost, determining the optimal *placement* of the sink nodes is a more difficult challenge. The initial deployment of the WSN can be done either in a structured or planned manner by a network designer, or in a semi-random way (e.g., an air-drop). In any case, the optimal placement of the sinks cannot be known a priori, and there is a need for heuristics to facilitate relocation of existing sinks or to position new sinks in the network. Our algorithms aim to find the optimal sink locations for a given network topology and coverage. The algorithms are employed at a separate computer and sink relocation is then performed either manually or by mobile sinks or robots.

Most works study *unconstrained* sink node placement, assuming that the sinks can be placed anywhere. In practice, there may be areas which are occupied by obstacles, or are out of wireless range, and therefore not viable for sink placement. Hence, in this paper, we study *constrained* sink node placement, meaning that the sinks can only be placed in a subset of the WSN scene. Via extensive simulations we show that the constrained approach leads to improved goodput and lifetime compared to the unconstrained approach.

Before presenting our own schemes, it is worth reviewing some of the preceding work regarding multiple sink deployment in WSNs.

II. Related work

Oyman et al. [1] propose to find the optimal placement of multiple sinks using the well-known K-means clustering. The cluster centroids for the k clusters are chosen as the optimal placement for the sinks. The approach is used to minimize the number of sinks for a predefined minimum operation period, and to find the minimum number of sinks while maximizing the network lifetime. The K-means method is further described and used as a baseline later in the paper. The approach presented in [1] requires global location information to find the optimal sink placements. Vincze et al. [2] aim to relax this requirement by approximating the location of nodes with unknown positions. The system is, however, based on a geographical routing protocol, which requires a functional location system in the WSN.

The approaches taken in [1], [2] study *unconstrained* sink placement. This limits their practical use. As discussed in the introduction, such schemes are based on the assumption that there are no physical boundaries limiting the proposed placement of the sinks. The presumed optimal sink locations found by the algorithms are therefore not necessarily viable in practice due to physical constraints in the scene. A proposed location may actually end up being outside radio-range of the surrounding sensor nodes. The work by Dai et al. [3] aims to solve this problem by only proposing sink positions at locations that are known to be in communication range with at least a subset of the network.

To accomplish this, they restrict sink placement only to locations already occupied by sensing nodes. However, since their network model is restricted to Manhattan grid layouts and assumes uniform link lengths and link weights, the approach is not useful for semi-structured deployments. In this sense, the works [4] and [5] are therefore considered more flexible. Although both works study relay node placement, they can be adapted to the sink node placement problem. Deployment constraints are used to limit relay node placements at some pre-specified candidate locations only, meaning that the proposed locations are not restricted to known sensor node locations as in [3]. Their methods are more flexible and practical in a real setting, but require that the deployment algorithm a priori knows the deployment constraints. This requirement cannot always be fulfilled.

The deployment strategies we present in this paper (SPP and RMP) distinguishes from the before-mentioned proposals since we allow *any* network topology. Also, sink deployment constraints are not an input parameter to the algorithms but are instead learned by inspecting the link information.

III. Sink Placement Algorithms

To effectively determine the optimal placement for multiple sinks, network information must be gathered globally or estimated. We distinguish the different schemes in two categories: (i) those that require knowledge about the geographical positions of all sensor nodes (geo-aware); and (ii), those that rely on the network topology (topology-aware). In the following, we present four different sink deployment strategies, two in each category. The first method is similar to the one previously proposed by Oyman et al. [1]. It also shares resemblance with the method proposed by Vincze et al. [2]. The tree final methods are considered novel to our paper.

A. K-means placement (KSP)

K-means is a classic and simple method for clustering that has been applied to several problem domains. When applied to sensor sink node placement, the cluster memberships proposed by the algorithm is ignored. K-means is simply used to find the cluster centroids given a set N of n sensor nodes and their geographical positions $P = \{p_1, p_2, \dots, p_n\}$. In this way, K-means can find the optimal set of sink locations $S^* = \{s_1, s_2, \dots, s_k\}$ given a predefined number of sinks k. The method works as follows:

1) The preferred number of sinks k is predetermined.
2) k points $s_1, \dots, s_k$ are placed in the geographical space bounded by the nodes being clustered, P. These points represent the cluster centroids, which will eventually constitute the sink locations.
3) Each sensor node is assigned to the cluster with the closest (Euclidean) centroid s.
4) The k centroids are repositioned to the mass center of each cluster.
5) Repeat steps 3-4 until the centroids no longer move.

By iteratively minimizing the within-cluster sum of squares, the final cluster centroids are found and chosen as the optimal placement for the sinks:
$S^* = arg\ min_S \sum_{i=1}^{k} \sum_{N_j \in S_i} ||p_j - s_i||^2$

The prerequisite to run K-means sink placement algorithm (KSP) is exact knowledge of each sensor node location. The location information can be obtained either by GPS positioning or by special localization schemes [6], [7]. In any case, the location information must be gathered from the sensor nodes to a central entity running KSP. This can be done using a mobile robot node or by temporarily installing one or more static sinks at random locations in the network.

B. K-medoid placement (KDP)

K-medoid clustering is closely related to K-means and is an excellent candidate algorithm for sink node localization. Instead of using cluster centroids, K-medoid builds on the concept of *medoids*. A medoid is defined as the most central object in a cluster. For our purpose, this is an attractive feature, since the algorithm can find the position of any k nodes in N that are most central instead of proposing *new* sink locations. The method therefore provide constrained placement, and our hypothesis is therefore that K-medoid is a better candidate for sink placement than K-means. Our K-medoid sink placement is based on Partitioning Around Medoids clustering (PAM), originally proposed by Kaufman and Rousseeuw [8]. The method works as follows:

1) Randomly select k of the n nodes to represent the initial medoids. The medoid positions will later represent the sink locations.
2) Each node is associated with the closest (Euclidean) medoid.
3) For each medoid m and non-medoid n, the pair (m,n) is swapped and the configuration cost is computed.
4) The configuration with the lowest cost is selected and stored in M.
5) Repeat steps 2-4 until there is no change in the medoid set.

The optimal sink locations are given by the positions of the medoid nodes in M^*, found by:
$M^* = arg\ min_M \sum_{i=1}^{n} min_{j=1}^{k} ||p_i - m_j||$

The above algorithm shares the same prerequisites as mentioned above for KSP, since all individual node locations must be known a priori.

C. Shortest path placement (SPP)

All multiple sink deployment strategies that require location information suffer from the following shortcomings:

1) The geographical positions of the sensor nodes must be known. To obtain the individual node positions, a localization and collection scheme must be present in the network.
2) Since the methods are based on Euclidean distance, the algorithms inherently assume that all sensor nodes share the same transmission range and that geographically adjacent nodes also are 1-hop neighbors. This is not always true in obstructed environments.

To overcome both these limitations, our Shortest Path Placement algorithm (SPP) can instead of requiring the geographical positions, take advantage of the network topology information to determine the optimal sink locations. By letting the sink placement algorithm take advantage of the topology information directly, instead of using the estimated node positions (which are imprecise and often derived from the topology anyway [6], [7]), the overall system design is radically simplified.

Our SPP algorithm builds on KDP and differs mainly in the distance measure employed. We model the network as an undirected graph G represented as a tuple $G(V, E)$ where V is the set of vertices representing the sensor nodes and E is the set of edges. Each edge represent a bidirectional communication channel between a pair of nodes i and j. We then construct an adjacency matrix A, where $a_{ij} = 1$ if there is an edge from vertex i to vertex j. If $i = j$, $a_{ij} = 0$. If there is no edge between i and j, $a_{ij} = \infty$. The all pairs shortest path matrix D is then computed from A using Dijkstras algorithm [9]. The shortest path distance between i and j is defined as d_{ij}. This measure now constitute the distance measure which replaces the Euclidian distance measure used in the KDP algorithm introduced above such that:

$M^* = arg\ min_M \sum_{i=1}^{n} min_{j=1}^{k} d_{ij}$

The algorithm finds k nodes (sinks) in the network that minimizes the average number of hops in respect to the remaining nodes in the network. The prerequisite to run SPP is that all links in the network are known a priori. As for the before-mentioned algorithms, such information can be gathered using a mobile node or by temporarily installing one or more sinks in the network. Notice that the collection of link information is inherently performed in many routing protocols, and this requirement is therefore easier to fulfill than obtaining the exact node positions.

D. Routing Metric placement (RMP)

Wireless sensor networks are error prone in nature and it is evident that poor link quality causes problems for packet delivery and routing. Hence, there are numerous works focusing on increasing the reliability by using better routing metrics, e.g., ETX, ETT or LQI. We provide an extension of the SPP algorithm that uses a *metric* for each edge before performing the shortest path calculation. The employed metric should preferably be the same metric as the one used by the routing protocol. The sink placement will then be optimized according to the chosen routing metric instead of being optimized to a separate (and often irrelevant) measure such as the Euclidean distance between the nodes.

As a proof-of-concept we use the link quality estimate (LQI) from 802.15.4 MAC layer to provide simple constraint based routing. The idea is implemented such that if the initial link quality estimate is below a certain threshold value (i.e., due to environmental constraints or path loss), we consider the link as *weak*. If the estimate is above this value, the link is considered *good*. By using this kind of routing constraint, the sink placement algorithm can be used to select the k sink node locations that maximize the overall link quality.

We extend the adjacency matrix A explained for SPP such that link constraints can be included in the calculations. This is implemented in the following manner:

$$a_{ij} = \begin{cases} 1 & \text{if link } i,j \text{ exists;} \\ 1+c & \text{if link } i,j \text{ is weak;} \\ 0 & \text{if } i=j; \\ \infty & \text{otherwise} \end{cases} \quad (1)$$

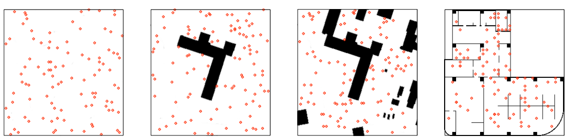
The constant c is used to take account for links which are considered weak. In our experiments, $c = 0.5$. The all pairs shortest path matrix D is computed from A, and inherently includes the link quality constraints. The shortest path distance between i and i is defined as d_{ij} and is used to find the sink locations as shown for SMP. RMP in this way finds the k nodes in the network that maximizes the average link quality. Placing the k sink nodes at these locations will presumably lead to fewer MAC retransmissions, fewer collisions and extended network lifetime.

IV. Anycast routing in multiple sink networks

In multiple sink WSNs, the sensor nodes usually transmits data to *one* arbitrary sink and do not particularly care which sink is used. In such an anycasting paradigm, the routing protocol is responsible for transmitting datagrams to at least one of the sinks that accept datagrams with a certain anycast address.

For the purpose of the studies in this paper, we have developed a tree based routing protocol. The protocol establishes an anycast collection tree routed at the sinks. All nodes transmit beacons indicating their distance to the sink, whereas sink nodes report a distance of 0. The protocol uses the link quality indicator (LQI) from the physical layer in addition to the hop distance in the routing decision. The LQI value of a link is measured upon beacon reception. If the LQI value is below a certain threshold value, the link is considered *weak*. The route cost then becomes a combination of the number of hops N_H and the number of weak links N_W. A route a is said to be better than route b if $N_W(a) < N_W(b)$ *or* $N_W(a) = N_W(b)$ *and* $N_H(a) \leq N_H(b)$. Thus, a data packet will follow the path that minimizes both the number of hops and the number of weak links between a node and a sink.

V. One-sink placement



(a) S1 (b) S2 (c) S3 (d) S4

Figure 1. The four scenarios used in the simulations

To obtain valuable understanding of the differences between our proposed deployment algorithms, we first study networks containing just one sink ($k = 1$). As a point of comparison for sink placement we use a simple center placement strategy. The strategy merely places the sink at the center of the area. Should the center position be blocked by an obstruction (i.e., wall or building), the sink is located at the nearest non-obstructed position. In this way, the model is supposed to mimic deployment as if performed by a physical network operator or a robot.

To ensure that our results are not biased by our selection of a particular network layout, we consider four different network scenarios as shown in Fig. 1. The first scenario represents an open area with no obstructions. The second scenario represents the same area but with a large obstruction (building). More buildings are added in the third scenario. The fourth scenario is an indoor office area. In all scenarios, we define that signals communicated through walls and buildings observe a different radio propagation condition than signal communication line-of-sight through open air. We use the ShadowingVis propagation model in ns-2.34 to model this behavior in the simulated areas.

For all scenarios, each sensor node transmits a 50-byte sensor reading packet each 100s addressed to the sink anycast address. The readings are transmitted during the entire lifetime of the network. We define the network lifetime as the point in time when the first sensor node runs out of energy. The simulation parameters, including the transmission and reception energy usage, are given in Table I. For simplicity we assume that the energy consumption during idle periods is negligible. All parameters are kept equal for the different deployment strategies, meaning that the only variable affecting the simulation results is the actual choice of sink deployment strategy. Initially, we place two sinks at two random locations. These sinks are used to collect neighbor information and link quality estimates, which are subsequently used in the calculations. For KSP and KDP, we assume that the geographical positions of the nodes are exact and known a priori.

Table I
Simulation parameters

Simulator	NS-2.34
Propagation model	ShadowingVis
pathlossExp_	1.5/4.0 (Open/Obstructed)
std_db_	2.0/1.9
dist0_	1.0/1.0
Number of nodes	100
Number of random topologies	10
Area	125m x 125m (S1-S3) 32m x 32m (S4)
MAC protocol	IEEE 802.15.4
Frequency	2.4 GHz
CSTresh_	1.20174e-07
RXThresh_	1.20174e-07
RXpower	35.28mW
TXpower	31.32mW
Initial Energy	1.0 Joule
Traffic parameters	CBR 50 bytes
Data rate	1pkt/100s/node

A. *Results and analysis*

Figure 2 show the lifetime for all scenarios and for all sink deployment algorithms. We observe that for scenario 1, the difference in lifetime is minimal between the five methods. This is expected considered that $S1$ represent a non-obstructed area, and with a reasonably high network density. For the scenarios $2-4$, we observe that the topology aware algorithms give remarkable lifetime improvements compared to both the geo-aware algorithms and the naïve center placement strategy. By concurrently studying Figure 2 and Figure 3, we observe that system lifetime relates to the average number of transmissions required to successfully transmit a packet from a source to the sink. This gives an insight of the quality of the links selected. Retransmissions due to packet loss cause more energy to be used on transmitting and receiving messages, which in turn reduces the system lifetime.

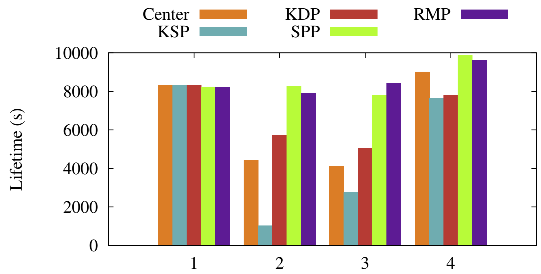


Figure 2. Network lifetime

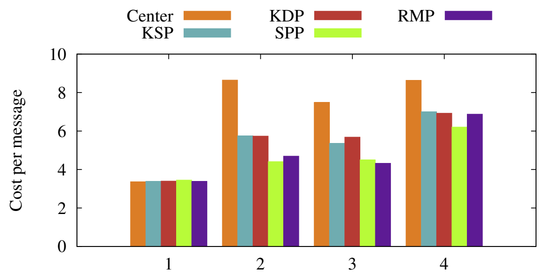


Figure 3. Average cost per sensor message

Figure 4 show the total number of sensor messages received at the sink (goodput) during the lifetime of the network. The figure in this way show the effective work performed by the sensor network during its system lifetime. KSP shows reduced performance for some topologies. This phenomenon is caused by the fact that the KSP strategy can propose sink locations in connection holes (no neighbors), or on top of obstructions. We did not reposition the sink to a better location in these cases. Center placement is therefore somewhat better, since obstructions are avoided in this model. However, there is still no guarantee that connection holes are avoided, and Center therefore has a lower average performance than the best deployment strategies.

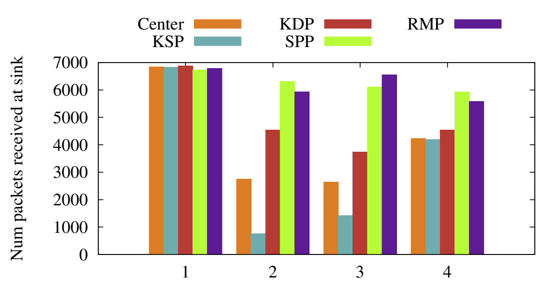


Figure 4. Total number of sensor messages received at the sink

Figure 5 show the percentage of nodes communicating with the sink during the system lifetime. This result gives a picture of how well the sink placement matches the network topology. Since all sensor nodes are randomly deployed within the open area, a small percentage of isolated nodes are expected regardless of the sink deployment procedure. However, the figure shows that an intelligent sink deployment procedure can minimize the number of isolated nodes. Again, we observe that the topology-aware strategies performs better that the other strategies.

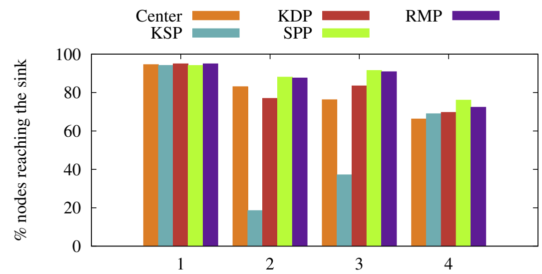


Figure 5. Percentage of nodes able to communicate with the sink

B. *Summary*

The following conclusions can be drawn from the above results:

- The network environment plays a huge part of the picture when comparing the performance of the schemes. When using a simple scenario ($S1$), all schemes give comparable results. However, in more complex environments which includes obstructions, SPP and RMP gives the longest lifetime, the highest number of packets received, and the lowest number of isolated nodes.
- RMP is the best choice when the network is sparse and there is a high number of low quality links in the network (i.e., many obstructions, as in $S3$). In a dense network ($S4$) and in a network with fewer obstructions ($S2$), SPP is the best choice. We anticipate that RMP may perform better under all network conditions if a more advanced network metric is used.
- We observe that even the simplest mechanism performs well under unconstrained and ideal conditions such as $S1$, while it performs poorly in obstructed environments. This result leads to the conclusion that previous sink deployment mechanisms only validated in simple simulation scenarios may be of little use in real world implementations.

VI. MULTIPLE SINK PLACEMENT

We now study the multi-sink problem and analyze the influence of increasing the number of sinks on the lifetime and total number of packets received. For the multi-sink case, we assume that the system does not particularly care which sink each sensor node uses as long as the lifetime is elongated and that the network load is balanced. We also assume that the sinks are either connected through a fixed network, or are manually collected by a network operator or robot after a certain period of time.

As the Center algorithm performed poorly for $k = 1$ and is difficult to apply for $k > 1$, we only consider the strategies KSP, KDP, SPP, and RMP. Also, we focus on scenario 3 only, since this scenario gave the results with the widest diversity for the different strategies in the one-sink case. We now investigate whether the difference between the strategies is consistent also when k increases. We apply the same simulation methods as described in Section V.

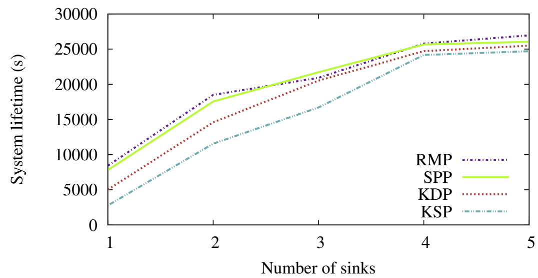


Figure 6. Lifetime of the sensor network

Figure 6 show the network lifetime related to the number of sinks for the different deployment strategies. We observe that the network lifetime first increases almost proportionally to the number of sinks, which is expected since the average path length decreases. It is also interestingly to see that the lifetime difference between the strategies observed for the one-sink case is sustained also when the number of sinks increases. This proves that it is extremely important to find the optimal sink placement even in the multi-sink case. It is, however, obvious that when a very high number of sinks is available (in this case $k \gg 5$), the choice of deployment strategy eventually becomes irrelevant. As in the one-sink case, we observe that the topology aware algorithms give remarkable lifetime improvements compared with the geo-aware algorithms. RMP increases the lifetime with 60% for $k = 2$, and 25% for $k = 3$ compared to KSP. In fact, *two sinks* deployed with SPP or RMP gives significantly longer lifetime than *tree sinks* deployed with KSP.

To get the full picture of how important it is to place the sinks wisely, Figure 6 must be seen in relation with Figure 7. Figure 7 shows the number of successfully received sensor readings at the sinks (goodput) during the system lifetime. We observe that with the topology-aware methods, SPP and RMP, the number of messages received during the system lifetime is significantly increased compared to the geo-aware methods, KSP and KDP.

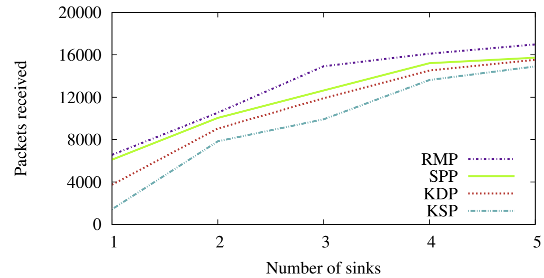


Figure 7. Total number of received packets

VII. Conclusions

In this paper, we have shown that deploying multiple sinks in WSNs offers a tremendous potential for improving both the lifetime and goodput. Most related work in the literature only considers *unconstrained* sink deployment mechanisms. Extensive simulation results show that such methods are insufficient since even the simplest deployment mechanisms performs well under unconstrained and ideal conditions, while they perform poorly in constrained environments. The results show that a constraint-based deployment algorithm is paramount to get the full potential of multiple sink WSNs.

References

[1] E. I. Oyman and C. Ersoy, "Multiple sink network design problem in large scale wireless sensor networks," in *In: Proc. of 1 st Int. Conf. on Communications (ICC 2004)*, 2004.

[2] Z. Vincze, R. Vida, and A. Vidacs, "Deploying multiple sinks in multi-hop wireless sensor networks," in *In: Pervasive Services, IEEE International Conference on*, July 2007, pp. 55–63.

[3] S. Dai, C. Tang, S. Qiao, K. Xu, H. Li, and J. Zhu, "Optimal Multiple Sink Nodes Deployment in Wireless Sensor Networks Based on Gene Expression Programming," in *ICCSN 2010*, February 2010, pp. 355–359.

[4] S. Misra, S. D. Hong, G. Xue, and J. Tang, "Constrained relay node placement in wireless sensor networks to meet connectivity and survivability requirements," in *In: IEEE INFOCOM 2008 Proceedings*, April 2008, pp. 281–285.

[5] D. Yang, S. Misra, X. Fang, G. Xue, and J. Zhang, "Two-Tiered Constrained Relay Node Placement in Wireless Sensor Networks: Efficient Approximations," in *In: IEEE Secon 2010 Proceedings*, June 2010, pp. 1–9.

[6] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, "Localization from mere connectivity," in *MOBIHOC*, 2003, pp. 201–212.

[7] J. Flathagen and R. Korsnes, "Localization In Wireless Sensor Networks Based on Ad hoc Routing and Evolutionary Computation," *MILCOM*, pp. 1173–1178, 2010.

[8] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley-Interscience, March 1990.

[9] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Num Math*, vol. 1, no. 1, pp. 269–271–271, Dec. 1959.