Localization in Wireless Sensor Networks based on Ad hoc Routing and Evolutionary Computation

Joakim Flathagen *‡, Reinert Korsnes *
* Norwegian Defence Research Establishment
‡ University Graduate Center Kjeller/NTNU
Email: {joakim.flathagen, reinert.korsnes}@ffi.no

***Abstract*—We propose evolutionary computation to estimate positions of nodes within a sensor network. The approach uses signal strength measurements between nodes and given positions for a subset of these nodes (anchor nodes). The signal strength measurements and routing requests take place simultaneously. A data collecting unit (sink node) receives distance estimates which are input to the evolutionary algorithm projecting node positions. This evolutionary approach can sort out data outliers and hence produce robust estimates of node positions. The present work contributes to decrease the cost and complexity of applying sensor networks. The approach also provides redundancy for the node positioning where alternative methods fail. The present simulations show examples of network generation and routing combined with estimation of node positions.**

***Index Terms*—Sensor Networks, Localization, Evolutionary Algorithm, DYMO-low**

I. INTRODUCTION

This work argues for evolutionary computation for localization within wireless sensor networks (WSNs). The approach provides low cost and robust localization utilizing signal strength measurements attached to routing control packets. A genetic algorithm [1] here searches for possible sets of node positions explaining these measurements. This search resembles the process of natural evolution.

Wireless sensor networks can consist of hundreds or even thousands of small sensing devices. Location awareness is crucial for many WSN applications such as environment monitoring and military surveillance. Sensor networks can also utilize location information for routing, cooperative computation, data fusion and location dependent sensor data requests [2], [3], [4], [5]. GPS positioning for every sensor node is not a general solution to localize nodes in sensor networks. It may be costly and impractical and sometimes irrelevant. Development of a precise, low-cost, reliable and fast converging localization scheme is therefore essential for the function of many sensor networks.

A. Related work

Localization schemes for sensor networks can be categorized depending on ranging, hardware, mobility, centralization and deployment restrictions. These will be briefly discussed here. *Range-independent* localization schemes [6], [7] determine node positions without using any special measurements. Localization is in this case a result of connectivity information. Even if *low cost hardware* can provide this capability, the position estimates are imprecise, especially for sparse networks. Additional information can improve the position estimates. Such additional information can be from measurements of distance or direction to known reference positions. These estimates are typically from measurements of time of arrival (TOA), time difference of arrival (TDOA), angle of arrival (AOA) or received signal strength indicator (RSSI). Examples of hardware for this type of measurements are ultrasound devices [8], angle-of-arrival antenna arrays [9] or laser [10]. However, introducing such additional hardware increases cost and complexity of sensor network systems.

The *deployment method* of sensor networks often determines the choice of its localization scheme. *Mobile* nodes can sometimes aid localization of individual sensor nodes [11], [12]. In such cases a vehicle, robot, or soldier enters the sensor field to assist the localization scheme. Sensor drop from airplanes, on the contrary, requires autonomous localization. *Centralized* methods may then be the only viable approach. Previous assumptions indicate that centralized methods are impractical due to high communication costs. Our proposal argues against this conclusion.

There are many previous attempts to estimate node positions in sensor networks using centralized search techniques. Kannan *et al.* [13], for example, creates an initial estimate of positions by applying simulated annealing and attempts to correct possible misplaced nodes thereafter. Tam *et al.* [14] apply evolutionary optimization to improve position estimates after initial triangulation, while Zhang *et al.* [15] more directly apply evolutionary computing for localization.

B. Our contribution

Most existing sensor network platforms can use signal strength measurements without additional hardware by employing RSSI from the IEEE 802.15.4 chipset. Our proposal takes advantage of such low-cost measurements to estimate node positions. However, other and more precise measurements (such as acoustic ranging) can be utilized if this is supported by the hardware. The proposal includes a prevalent WSN ad hoc routing protocol, DYMO-low [16] that is exploited to fetch and distribute RSSI values. By combining route establishment and localization our approach contributes in reducing the effort and complexity of sensor network deployments. The sink employs an *evolutionary approach* to provide estimation of node positions using the information gathered. This gives a reasonable robust solution even for poor signal strength measurements. Our implementation of an evolutionary algorithm is simple and intuitive and relaxes the search space as compared to similar work [14], [15]. Section III below clarifies this relaxation via variation of fitness measures and allowing for data outliers.

The remainder of this paper is organized as follows. Section II describes collection and distribution of RSSI measurements using DYMO-low. Section III elaborates the evolutionary localization algorithm. Section IV presents simulations results and analysis. Section V gives concluding remarks.

II. Simultaneous routing and RSSI measurement

The support of RSSI measurements is common in prevalent IEEE 802.15.4 implementations. But signal strength measurements, especially indoors, may provide imprecise distance estimates due to multipath propagation, reflection and channel fading. RSSI measurements are therefore mainly ignored for node positioning within sensor networks. However, recent research by Holland *et al.* [17] shows that RSSI measurements on sensor nodes strongly correlate with distance. RSSI measurements on a link are also symmetric [18] and localization schemes for small-scale networks can utilize this property [19].

Our scheme extends this work by using RSSI to aid localization in medium to large-scale multihop networks. We have chosen a centralized approach to localization, meaning that only the sink is involved in computation of the node positions. It is worth noting that distributed protocol designs are traditionally preferred before centralized designs in networking systems due to the fault tolerance and lack of scalability of the latter approach. However, we argue that in most WSNs, the sink node is already a single point of failure, and the fault tolerance is not increased by centralizing the localization algorithm. In fact, it simplifies the protocol design and its implementation. Further, the scalability of the centralized algorithm is not a big concern compared to a distributed design, as the sink node can be equipped with several orders of magnitude more memory and CPU than the sensor nodes.

The approach in this paper exploits route establishment to perform measurements and transport RSSI values to the sink by extending the reactive distance vector routing protocol DYMO-low [16]. DYMO-low is intended for use on IEEE 802.15.4 devices and is based on the principle of flooding route requests (RREQ) and unicasting route replies (RREP) as known from AODV [20] and DYMO [21]. Our extension introduces two new messages to the protocol, Localization Route Request (LRREQ) and Localization Route Reply (LRREP). Fig. 1 illustrates the protocol operation consisting of a request phase and a reply phase.

A. Request phase

The sink initiates the network by announcing its address via a Localization Route Request (LRREQ). This message can be seen as a proactive route request destined to *all* nodes in the network. The LRREQ is flooded similarly as a regular DYMO-low routing request (RREQ). The nodes which receive

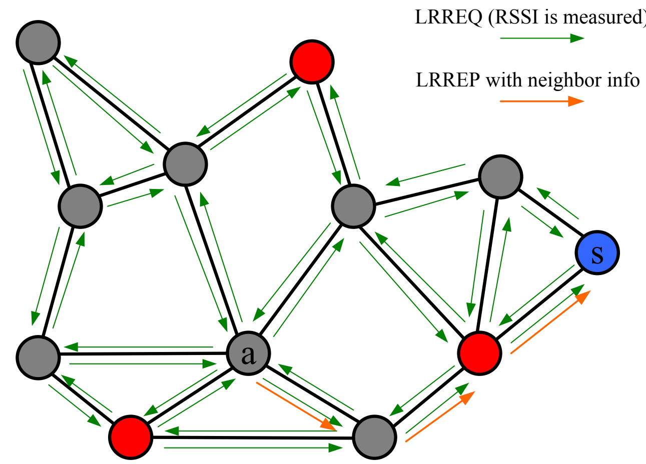

Fig. 1. Initialization of sensor network with anchor nodes (red). The sink S (blue) starts route discovery. All nodes collect RSSI measurements from their neighbors while the LRREQ disseminates. They report their measurements as attachments to their individual route reply back to the sink. Anchor nodes also report their positions. The sink then estimates node positions using evolutionary computation.

the LRREQ, retransmit the packet only once. This means that all nodes will receive a copy of the LRREQ from each of its neighbors (cf Fig. 1). When a node receives a LRREQ packet, it performs a RSSI measurement and subsequently stores its value and the address of the sender. As the LRREQ disseminates from the sink to the entire network, all nodes will eventually obtain a distance estimate to each of its one-hop neighbors with no more cost than a regular Route Request.

B. Reply phase

A node will, after receiving a packet, respond back to the sink using a LRREP (Localization Route Reply). This transmission takes place after a random time delay to avoid network congestion and collisions. Fig. 1 illustrates this process. The response message extends the regular Route Reply defined in DYMO-low, with a list of the one-hop neighbors and their correspondent RSSI measurement values. Anchor nodes will also add their own present position that can be from a GPS receiver. The sink will eventually receive LRREPs from all the sensor nodes in the network. It will then use this information to estimate the individual locations.

Note that the sink receives two RSSI measurement values for each link in the network (one from each link end). The distance estimation applies the mean of the values from each link. The duplicated information also enables reconstruction of missing LRREP information. If the LRREP from for example node a in Fig. 1 is lost on its way to the sink due to congestion or collision, the sink can use the LRREPs from the surrounding neighbors of a to estimate its location. This gives a minimum level of redundancy.

C. Features and considerations

The above approach provides two important additional features.

i) After a complete request/reply phase, all nodes in the network has a valid route to the sink, making them ready to perform their sensing task immediately. Notice that if standard DYMO-low is used, route requests must be initiated from each node in the network to accomplish this. This could cause tremendous overhead due to the flooded route requests. Our approach, on the contrary, limits this to just *one* sink-initiated route request and considerably reduces the number of messages flooded in the network.

ii) By using the information provided in the LRREPs, the network operator at the sink will know which of the nodes in the network are fully functional, within range and operating. It could later be useful to add other sensor information to the LRREP message in order to inform the sink that the individual sensors on each node are operating satisfactorily after deployment.

The size of the LRREP may end up being too large for a IEEE 802.15.4 frame if i) a node has a very large number of one-hop neighbors, or ii) the LRREP includes much status information from the sensors on the node, or iii) a combination of the two. The LRREP will still be transmitted, as the 6LoWPAN sublayer [22] elegantly fragments and reassembles datagrams being larger than a MAC-frame. However, in sparse and medium density networks fragmentation of the LRREP are not likely to occur.

III. Localization through evolutionary computation

This section describes our centralized evolutionary computation method to estimate node positions within the sensor network. Evolutionary approaches generally provide capacities for searching through large sets of possible explanations of given data. For our purpose, such techniques are therefore particularly interesting for estimation of node positions from error-prone data, such as signal strength measurements.

Parameter estimation is often equivalent to model identification from data. A set of parameter values then typically defines a model within for example a physical setting. The actual set of parameters is in our case the (unknown) set of sensor node positions, and the data is the distance estimates from the RSSI measurements.

Given a set $\boldsymbol{I} = \{\boldsymbol{N}_1, \boldsymbol{N}_2, \ldots, \boldsymbol{N}_K\}$ of K nodes and estimates (measurements) of the distance between them. If the distance between two nodes is within a common detection range r, the estimate is assumed to be a result from measurements (explicit detection). Otherwise, the estimate only tells that the distance between them is larger than r (i.e. missing data defaults to an implicit imprecise distance estimate larger than r). Our algorithm utilizes information inherent in missing or negative observations.

For $i = 1, \ldots, K$, let the real position vector $\boldsymbol{r}_i$ denote an initially proposed location for the node $\boldsymbol{N}_i$. These positions can be restricted to for example a rectangular (test) area covering the whole sensor network. The present simulations are for a rectangular test area of size $(100 \times 50\,\mathrm{m}^2)$. Some nodes acting as anchor points have known location, while the position vectors for the other nodes are random.

Anchor nodes with known positions give the possibility to estimate all node positions from internode distance data. The positions $\boldsymbol{r}_i$, $i = 1, \ldots, K$, constitute a proposal explaining the inter-node signal strength data. They also constitute "genes" in the present setting. The algorithm generates a population $\mathcal{P}$ of $L = 1000$ such proposals for node positions. A fitness measure $\boldsymbol{m_f}$ quantifies how well each proposal (individual) in $\mathcal{P}$ fits to the measurement data. The fitness measure provides a linear ordering in the population defining for each individual its probability for producing offspring.

The algorithm generates an offspring by randomly selecting two individuals (parents) in the population $\mathcal{P}$. The position $\boldsymbol{r}_i$ of node $\boldsymbol{N}_i$ for the offspring is then a copy of the corresponding position for one of the parents with equal (50-50) probability. A random mutation with probability 0.02 takes place as a random displacement $\Delta\boldsymbol{r}$ relative to this position. A mutation may be small or large. In our example simulations we apply three types of mutations in alternating sequence during the generations of the evolutionary process. One type of mutations are large mutations $\Delta\boldsymbol{r}$ with a uniform distribution over set $[-100, 100] \times [-50, 50]\,\mathrm{m}^2$ (i.e. $\Delta\boldsymbol{r} \in [-100, 100] \times [-50, 50]$). The two other types of mutations are similarly for $\Delta\boldsymbol{r} \in [-10, 10] \times [-5, 5]$) and $\Delta\boldsymbol{r} \in [-1, 1] \times [-0.5, 0.5]$). Mutations only take place if they result in a new position inside the rectangular test area in which the sensor network is known to be.

The evolutionary process takes place in cycles where a number of $S = 100$ of the best fit individuals survive through *elitism* and the remaining $L - S = 900$ exits the population. For each cycle, 900 new individuals are created. The new population of $L = 1000$ proposals constitute the new generation.

Let $o_{i,j}$ denote the estimate of the distance between the nodes $\boldsymbol{N}_i$ and $\boldsymbol{N}_j$ $(i, j = 1, 2, \ldots, K)$. The statement $o_{i,j} > r$ is equivalent to no available data despite of good attempts to detect. Assume

$$s_{i,j} \stackrel{def}{=} |\boldsymbol{r}_i - \boldsymbol{r}_j| \tag{1}$$

is the distance derived from the model sensor network $\boldsymbol{I} \in \mathcal{P}$. A possible fitness measure $\boldsymbol{m}^p_{\boldsymbol{f}}(\boldsymbol{I})$ for an individual $\boldsymbol{I} \in \mathcal{P}$ can here be a power sum of the differences between observed distances and the distances according to $\boldsymbol{I}$:

$$\boldsymbol{m}^p_{\boldsymbol{f}} \stackrel{def}{=} \sum_{i,j=1}^{K} d^p_{i,j} \tag{2}$$

where

$$d^p_{i,j} = \begin{cases} |s_{i,j} - o_{i,j}|^p & \text{if } s_{i,j} \leq r \text{ and } o_{i,j} \leq r \ ; \\ 0 & \text{if } d_{i,j} > r \text{ and } o_{i,j} > r \ ; \\ |2r|^p & \text{otherwise} \end{cases} \tag{3}$$

The algorithm applies the fitness measure $\boldsymbol{m}^p_{\boldsymbol{f}}$ for $p = 1, 2$. Variation of the fitness measure $\boldsymbol{m_f}$ during the evolutionary process extends the search space making the evolution less likely to stagnate at local optima.

Note that the evolutionary method above extends the search space proposed by Zhang *et al.* [15] which restricts possible node positions to conform to observed neighborhoods.

Evolutionary computation may function in a setting with frequent occurrence of data outliers. The above approach is directly extendable so it can perform combined analysis of spatially related sensor data and sensor positioning. This can extend the application space to for example data transmission ranges shorter than otherwise applicable for sensor node positioning. Note that an evolutionary approach also has possible parallel implementations. This gives the opportunity to distribute computational load available in the sensor network.

IV. Test and Evaluation

The most challenging scenario for a localization scheme is when the nodes are randomly deployed, such as during an airdrop. The example scenarios below are therefore for such situations. In randomly deployed networks, the network degree defines whether the nodes are uniquely localizable or not. The routing scheme is also sensitive to the degree of network connectivity. These aspects are studied in the following simulations.

Two randomly deployed scenarios illustrate the evolutionary localization algorithm. The first scenario is for ideal RSSI measurement conditions (zero measurement error). The second scenario is for more realistic RSSI measurement errors and inaccuracies found using present implementations of IEEE 802.15.4.

A. Network impact

As no simulator implementation of DYMO-low was public available, we implemented the Internet Draft in the NS-2.34 network simulator [23]. Then the proposed extensions to the protocol were added to enable measurement and distribution of signal strength.

In the simulations, the packet overhead involved in performing a complete routing and localization process was studied. The routing scheme was evaluated under the effect of network density and node population. The setup used the IEEE 802.15.4 MAC layer and 20 different random simulation topologies were run for each setup.

1) Localizable nodes: The fraction of localizable nodes was examined for different network densities. Given the area A, the number of nodes K and the radio range r, then the average number of possible neighbors d is defined by the average number of nodes within the area $((\pi r^2 K)/A)$. This measure does not account for area edge effects. A node at a corner of a rectangular area will only have an average of $d/4$ neighbors. A density d of 5 here represents a sparse network and a density of 20 a dense network. For each density simulated, the number of nodes K was varied between 50 and 200. The results are shown in Fig. 2.

In a sparse network ($d = 5$) only 70% of the nodes could be localized by the sink, meaning that 30% of the network was partitioned. When $d = 7$, more than 90% of the nodes could be localized. This increased to about 100% when $d = 20$.

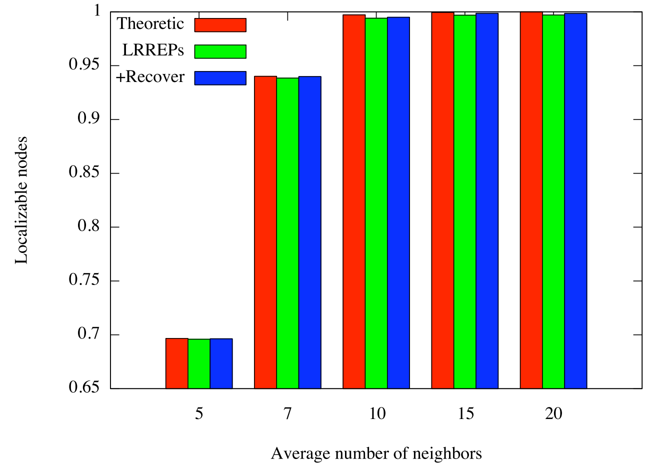

Fig. 2. Fraction of nodes localizable for different network densities. Red bars show theoretical fraction of nodes in the network reachable by the sink node. Green bars show fraction of nodes reachable using incoming LRREPs. Blue bars show fraction of nodes reachable by reconstructing missing LRREPs.

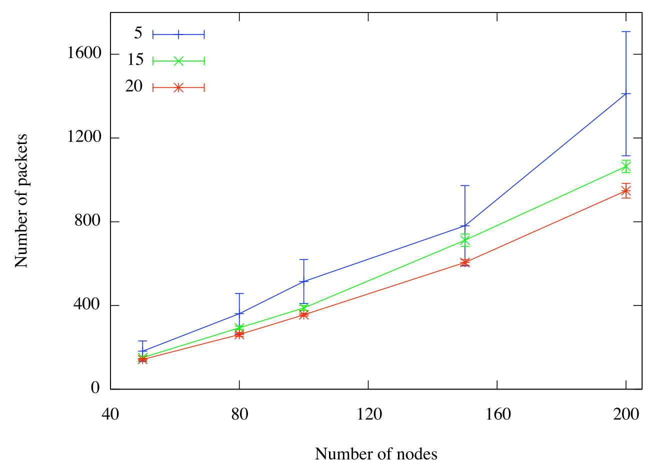

Fig. 3. Number of transmitted packets as a function of network size (number of nodes) for complete route discovery including RSSI-measurement. The average number of neighbors varies between 5–20. The 95% confidence interval included.

A small number of route replies (LRREPs) was lost due to collisions or congestion in the network. This caused the actual number of localizable nodes to be lower than the theoretic, as shown in the green bars. The protocol was however, able to reconstruct 60–70% of this lost information thanks to the redundant information in other LRREPs.

2) Network overhead: Fig. 3 shows the total number of packets transmitted to obtain RSSI measurements and route discovery in randomly deployed networks. The number of packets increased with increasing number of nodes. It also increased with lower density due to more hops between an arbitrary node and the sink.

Fig. 4 represents the same network topologies as for Fig. 3 while quantifying data transport in terms of number of bytes instead of number of packets. The total number of bytes

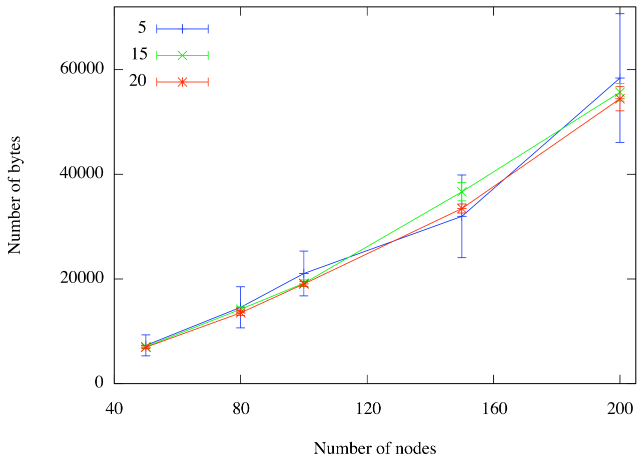

Fig. 4. Number of transmitted bytes as a function of network size (number of nodes) for complete route discovery including RSSI-measurement. The average number of neighbors varies between 5–20. The 95% confidence interval included.

transmitted was approximately constant for a given number of nodes regardless of the density. There is in this way a balance between a tendency for increased traffic due to decreased number of hops and an increase due to larger data packets caused by more one-hop neighbors.

The scheme seems to scale well and we state that the data requirement to run the scheme is within the limits of IEEE 802.15.4.

B. Localization performance and accuracy

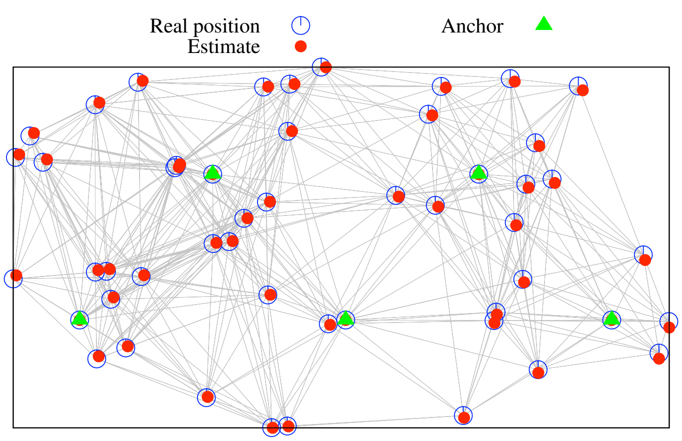

Fig. 5. Result from numerical experiment with 50 nodes including 5 anchor nodes and no measurement errors.

A separate and simple Ada program implemented the proposed evolutionary algorithm. The algorithm was evaluated under the effect of RSSI measurement quality.

As identified in Fig. 2, randomly deployed nodes require a high node density to avoid network partitioning. Therefore we have considered an initial setup consisting of 50 nodes within a $100 \times 50\,\mathrm{m}^2$ rectangular area and with transmission range $r = 30\,\mathrm{m}$. Fig. 5 is for a simulation with ideal measurements (no measurement errors). The average position error is in this case within 0.5 m. However, this error is an artificial effect by the model since candidate solutions (individuals) in the evolutionary process can be subject to fine tuning with arbitrary small mutations. Note that the scenario cannot be considered realistic unless the conditions are ideal or a more exact measure than RSSI is employed. Fig. 6 illustrates a typical generational development of the fitness $\boldsymbol{m}_{\boldsymbol{f}}^{p}(\boldsymbol{I})$ for the most fit individual $\boldsymbol{I}$ in the population in this scenario. The fitness measure $\boldsymbol{m}_{\boldsymbol{f}}^{p}$ did here drive the evolutionary process where the value of p changed between 1 and 2 for each 5 generation.

Fig. 7 shows results from a simulation with significant measurement errors possessing a uniform distribution around the real distance ± 10 percent. Solid black lines here illustrate data outliers which are 50 percent less than the real distance. Such measurement errors are here similar to real sensor nodes [17]. The final position estimates show small errors (average less than 1 m).

The performance of the evolutionary algorithm is sensitive to reduction of the transmission range r or the network degree (or node density) and the spatial distribution of the anchor nodes.

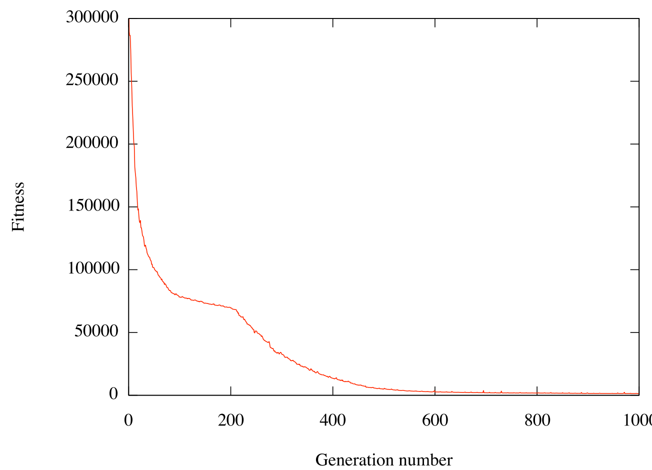

Fig. 6. A typical generational development of fitness by the evolutionary algorithm.

C. Summary

We have contributed to the discussion of applying evolutionary computation to estimate positions of nodes. Evolutionary computation seems to provide simple solutions to complex data fusion tasks. Our example simulations indicate that current hardware and standards may provide possible pioneering attempts in this direction. The provided simulation results also show that the data requirement to run the localization scheme is well within the limits of IEEE 802.15.4, meaning that centralized localization is feasible.

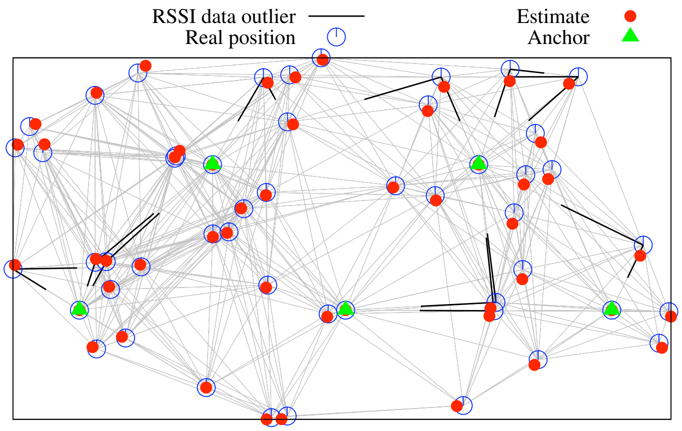

Fig. 7. Result from numerical experiment with 50 nodes including 5 anchor nodes. The error has in this case an uniform distribution around the distance ± 10 percent. Solid black lines illustrate data outliers which are 50 percent less than the real distance.

V. Conclusion

We argue that both the ranging measurements, the measurement data gathering and the localization algorithm are essential in providing a complete localization system solution. In this paper a scheme including all those components is presented. The proposed localization scheme is based on centralized evolutionary computing and employs the route establishment phase of DYMO-low to fetch and distribute signal strength values.

We conclude by emphasizing the flexibility in the scheme presented in this paper. The proposed extension to the DYMO-low protocol can potentially be used to facilitate other centralized localization algorithms than the evolutionary computation algorithm proposed here. Likewise, the evolutionary algorithm can take advantage of information gathered using a link state routing protocol, such as OLSR [24]. Further, the evolutionary algorithm can benefit from more precise ranging methods such as acoustic ranging. This makes our contributions versatile and attractive to a wide range of WSN applications.

References

[1] W. Banzhaf, P. Nordin, R. E. Keller, and F. D. Francone, *Genetic Programming – An Introduction; On the Automatic Evolution of Computer Programs and its Applications*. San Francisco, CA, USA: Morgan Kaufmann, Jan. 1998.

[2] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, no. 3, pp. 325 – 349, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/B7576-4B3JKH2-3/2/0baf704778ae8cf6e3651eb9f329e4c0

[3] S. Chen, G. Fan, and J.-H. Cui, "Avoid 'void' in geographic routing for data aggregation in sensor networks," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 1, no. 4, pp. 169–178, 2006.

[4] D. Espes and Z. Mammeri, "Backbone-based location-aided routing algorithm to reduce control packets of AODV protocol," in *Mobility '06: Proceedings of the 3rd international conference on Mobile technology, applications systems*. New York, NY, USA: ACM, 2006, p. 15.

[5] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker, "GHT: a geographic hash table for data-centric storage," in *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. New York, NY, USA: ACM, 2002, pp. 78–87.

[6] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, "Localization from mere connectivity," in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*. ACM Press New York, NY, USA, 2003, pp. 201–212.

[7] D. A. Tran and T. Nguyen, "Localization In Wireless Sensor Networks Based on Support Vector Machines," *IEEE Trans. Parallel Distrib. Syst.*, vol. 19, no. 7, pp. 981–994, 2008.

[8] L. Girod and D. Estrin, "Robust range estimation using acoustic and multimodal sensing," *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, vol. 3, pp. 1312–1320 vol.3, August 2002. [Online]. Available: http://dx.doi.org/10.1109/IROS.2001.977164

[9] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AOA," *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 3, pp. 1734–1743 vol.3, April 2003.

[10] R. Farrell, R. Garcia, D. Lucarelli, A. Terzis, and I.-J. Wang, "Localization in Multi-Modal Sensor Networks," *Intelligent Sensors, Sensor Networks and Information, 2007. ISSNIP 2007. 3rd International Conference on*, pp. 37–42, December 2007. [Online]. Available: http://dx.doi.org/10.1109/ISSNIP.2007.4496816

[11] M. L. Sichitiu and V. Ramadurai, "Localization of wireless sensor networks with a mobile beacon," *Mobile Ad-hoc and Sensor Systems, 2004 IEEE International Conference on*, pp. 174–183, October 2004.

[12] S. Lee, E. Kim, C. Kim, and K. Kim, "Localization with a mobile beacon based on geometric constraints in wireless sensor networks," *Wireless Communications, IEEE Transactions on*, vol. 8, no. 12, pp. 5801–5805, December 2009. [Online]. Available: http://dx.doi.org/10.1109/TWC.2009.12.090319

[13] A. Kannan, G. Mao, and B. Vucetic, "Simulated Annealing based Wireless Sensor Network Localization," *Journal of Computers*, vol. 1, no. 2, 2006.

[14] V. Tam, K. Cheng, and K. Lui, "Using micro-genetic algorithms to improve localization in wireless sensor networks," *J. of Comm., Academy Publisher*, vol. 1, 2006.

[15] Q. Zhang, J. Wang, C. Jin, J. Ye, C. Ma, and W. Zhang, "Genetic Algorithm Based Wireless Sensor Network Localization," *Natural Computation, 2008. ICNC '08. Fourth International Conference on*, vol. 1, pp. 608 –613, oct. 2008.

[16] K. Kim, G. Montenegro, S. Park, I. Chakeres, and C. Perkins, "Dynamic MANET On-demand for 6LoWPAN (DYMO-low) Routing," Internet Engineering Task Force, Internet-Draft, Jun. 2007, work in progress. [Online]. Available: http://www.ietf.org/internet-drafts/draft-montenegro-6lowpan-dymo-low-routing-03.txt

[17] M. Holland, R. Aures, and W. Heinzelman, "Experimental investigation of radio performance in wireless sensor networks," *Wireless Mesh Networks, 2006. WiMesh 2006. 2nd IEEE Workshop on*, pp. 140 –150, sept. 2006.

[18] K. Srinivasan and P. Levis, "RSSI is Under Appreciated," in *In Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets*, 2006.

[19] A. Awad, T. Frunzke, and F. Dressier, "Adaptive Distance Estimation and Localization in WSN using RSSI Measures," *Digital System Design Architectures, Methods and Tools, 2007. DSD 2007. 10th Euromicro Conference on*, pp. 471 –478, aug. 2007.

[20] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561 (Experimental), Jul. 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3561.txt

[21] I. Chakeres and C. Perkins, "Dynamic MANET On-demand (DYMO) Routing," Internet Engineering Task Force, Internet-Draft draft-ietf-manet-dymo-19, Sep. 2009, work in progress. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-manet-dymo-19.txt

[22] G. Mulligan, "The 6LoWPAN architecture," in *EmNets '07: Proceedings of the 4th workshop on Embedded networked sensors*. New York, NY, USA: ACM, 2007, pp. 78–82.

[23] Software, "The ns-2 network simulator," Accessed 2010. [Online]. Available: http://www.nsnam.org

[24] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," RFC 3626 (Experimental), Oct. 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3626.txt