Localization in Wireless Sensor Networks by Constrained Simultaneous Perturbation Stochastic Approximation Technique

Mohammad Abdul Azim*, Zeyar Aung*, Weidong Xiao*, Vinod Khadkikar* and Abbas Jamalipour†
* Masdar Institute of Science and Technology
Abu Dhabi, United Arab Emirates
{mazim, zaung, mwxiao, vkhadkikar}@masdar.ac.ae
† School of Electrical and Information Engineering
University of Sydney
Sydney, Australia
abbas.jamalipour@sydney.edu.au

***Abstract*—Localization of sensor networks poses an immense challenge and is considered as a hot research topic in recent days. To address the accuracy on localization this paper proposes constrained simultaneous perturbation stochastic approximation (SPSA) based localization techniques for wireless sensor networks. A simple centralized localization algorithm using SPSA technique that estimates the location of the non-anchor nodes based on minimizing the summation of the estimated error of all neighbors is the basic building block of the proposed localization technique. This category of localization technique incurs errors often referred as flip ambiguity. The improvement of the simple SPSA based localization is made by modifying the algorithm to a constrained optimization technique using penalty function method where the correction on the flipped node is made by penalizing the identified flips by the penalty function. Simulation results demonstrate the superiority of the proposed SPSA algorithm compared to its closest counterpart, namely, the simulated annealing (SA) based localization algorithm.**

***Index Terms*—wireless sensor network, localization, constrained optimization, simultaneous perturbation stochastic approximation**

I. Introduction

Diverse applications can be developed and deployed by using wireless sensor networks (WSNs) to satisfy the needs of different kinds of users. Regardless of the use and type of sensor applications, the sensed or measured value is beneficial only if the location of sensor is available. Thus, localization is an active area of research in the context of WSN.

A. Background and motivation

Localization in WSN poses a challenge primarily due to the device cost, resource constraints and poor accuracy.

Straight forward solution to localization through GPS is not very suitable because of the device cost and size. GPS is power hungry and it necessitates line of sight from the GPS satellites. Consequently GPS free solution to the localization becomes a necessity for WSN localization. As a result a number of such techniques are developed and presented in the context of WSN. In fact the localization algorithms use a subset of sensors known as anchors where the location information is known. With the help of the anchors and based on the distance and/or angle measurements the algorithms derive the location information of non-anchor sensor nodes. Localization problem remains an open issue and a hot research topic due to the challenges posed by large error and high cost of the localization techniques.

Localization techniques largely follow stages of measuring distances or angles of neighboring nodes and combining them to derive the relative or exact location of the nodes in the networks. Three fundamental methods of taking the aforementioned measurements are receiving signal strength indicator (RSSI) [26] based, time of arrival (ToA) [14][8]/ time difference of arrival (TDoA) [25][10] based and angle of arrival (AoA) [23] based techniques.

RSSI uses signal propagation loss as its underlying idea. First the transmitter transmits with predefined signal strength or embeds the signal strength with the hello message. Upon receiving, receiver measures the signal strength and finds the propagation loss by using theoretical or empirical models. This propagation loss can be translated into a distance estimate. In a ToA/TDoA approach the propagation time of the signal (from transmitter to receiver) can be translated directly into distance using signal propagation speed. Where an AoA technique can measure the angle at which the signals are arrived. These measurements are then used to find the location by using simple triangulation techniques. Various estimation techniques can be used to improve the error incurred by the measurements.

Unfortunately each of the aforementioned techniques has its own limitations. RSSI is a cheap solution with respect to the hardware cost. But due to the unreliability and randomness of the wireless propagation RSSI suffers from its poor accuracy, especially due to the multipath propagation. TDoA provides good result only if there exists a line of sight condition where this condition is not practically achievable in many deployments. AoA can provide a reasonably accurate measurement with a high hardware cost. In fact an array of receiver is required to measure the angle information where the delay of arrival at each element is measured and converted to AoA. A higher accuracy of angle measurement requires a higher number of receivers in the receiver array results higher hardware cost.

B. Our contributions

To address the localization problem in sensor networks this paper proposes a constrained SPSA based algorithm in WSN. The contributions of the paper are twofold:

1) Utilizing the simultaneous perturbation stochastic approximation (SPSA) algorithm [29] as a tool to approximate the locations by minimizing localization error based on a specific cost function.
2) Handling flip ambiguity by incorporating a constrained problem where the constrain is addressed by a penalty function method (logarithmic).

The rest of the paper organized as follows. Section II presents a background of such localization algorithms in the literature, section III presents and defines the cost function along with the SPSA localization algorithm. Section IV presents simulation results in various perspectives. Finally section V concludes the paper along with some future directions of this particular approach.

II. Related works

A number of localization algorithms are presented in the literature [4][21]. Some of them are simple, lightweight but suffers from high error in deriving the locations. Among these coarse-grained techniques, reference [7] proposes the estimated location of sensor node be the centroid of the location of the neighboring anchors. The error performance of the localization technique is improved by incorporating a weight attached to each neighbors and the resultant location is simply the weighted average of the neighbors [6][31]. A further improvement of the localization algorithm is made by adaptively determining the relative weights to the neighbor nodes [5].

Another lightweight localization algorithm is DV-hop [20]. The DV-hop first finds the hop distances of the nodes from the anchors by incorporating distance vector routing technique. The hop count is then translated to the actual distance by determining the average hop distance in meters based on the actual distance of the anchors and their hop distance counts. RSSI DV-hop (RDV) improves the simple DV-hop performance by replacing the hop count to the RSSI based distance measurements [30].

In reference [2] authors take average of RSSI values with different transmit power levels to construct a table. The table is processed centrally to compensate the non-linearity and thereby estimate the distances between nodes. Using sequential quadratic programming method the final results are achieved by minimizing the cost function.

A multidimensional scaling (MDS) based centralized localization is proposed in [28]. Algorithm first determines the shortest paths of all node pairs which can easily be determined by the Dijkstra's [11] or Floyd's [13] algorithm. The distances then assigned as the elements of distance matrix of MDS. A relative map of the nodes is obtained by the classical MDS from the MDS matrix. Based on the positions of the anchor nodes an absolute map is derived from the relative map.

By collaborative multilateration nodes several hops away from the beacon enable anchors can derive their locations [27]. Nodes find the bounding box or the region where it lies based on the beacon coordinates. Nodes use Kalman filtering approach to update the positions. Where nodes not directly connected to the beacons starts with the neighbors as a reference point. After calculating nodes broadcast the location estimates. And upon receiving the estimates neighbors updates their own estimates. By using this iterative approach location information becomes refined throughout the network. The drawback of the system is, it requires updating of location through transmissions. Though distributed, the transmission in each round is energy hungry.

Algorithm in reference [24] assumes nodes as point masses and are connected through springs. Algorithm uses force directed relaxation method to converge to a minimum energy configuration. This heuristic graph embedding method uses a polar coordinate approach to the localization algorithm. The drawback is, the algorithm is vulnerable to stuck into local minima.

In [3] nodes in the network that have the global location information (known as seed) initiate the outward broadcasting of hello messages that contain the seed location information and hop count from the seed. Intermediate node suppresses the duplicate messages and finds the minimum hop counts from the seeds. Upon finding such three different hop counts and seed location information nodes calculate their positions by finding the minimum of total squared error between calculated and estimated distances. The algorithm requires high node density to keep the localization error reasonable. Moreover using hop count can incur error if the hello message goes through a detour due to obstacles.

Algorithm [22] uses maximum likelihood (ML) estimation for location estimation of sensor node by minimizing the difference between measured and estimated distances. Minimum mean square error (MMSE) [17] algorithm is used to solve the aforementioned estimation problem. The drawback of the technique is it requires a high density of node, alternately a high transmit range to make the estimation from a reasonably large neighborhood cluster. With a small number of neighbor nodes this algorithm suffers from a poor performance [9][19].

In a similar approach simulated annealing (SA) based localization [16] solves the minimization problem with simulated annealing technique. Algorithm is evaluated with at least three anchors in the neighborhood for all non-anchor nodes deemed an impractical deployment for the randomly deployed sensor networks. Moreover the algorithm suffers from phenomena called flip ambiguity [18][12][15] addressed later in the paper.

III. CONSTRAINED SPSA LOCALIZATION

The proposed localization algorithm randomly initializes the locations of the non-anchor nodes in the networks. It then estimates the location information based on the measured distances of the neighbor nodes. The estimation is based on the technique known as SPSA that minimizes the sum of error between the measured and estimated distances of the neighbor nodes. In fact the aforementioned cost is modified to incorporate penalty for the flipped nodes by using a penalty function method.

A. Collecting measurements

Algorithm starts with measuring distances between all neighbor nodes. The simplest and cheapest way of taking such measurement by transmit-receive signal strengths can be chosen. Where the distance can be measured as $d_{ij} = \sqrt[\beta]{p_i/p_j}$. Here p_i and p_j are transmit and receive signal strength respectively and β is the path loss exponent and can be measured at unit distance. Unfortunately these straight forward measurements incorporate errors due to the nature of wireless medium and can be modeled by log-normal shadowing [1]. According to the model, the receive signal varies as $[\mu, \sigma^2]$ where μ and σ are mean and variance and often taken as 0.0 and 1.0 respectively. The derived measurements along with the node IDs are then sent to the central location for further processing where location estimations are derived through multiple refinements. An alternative distance measurement by using ToA/TDoA can also be chosen.

B. The basic cost function

One most important challenge of WSN localization comes from the fact that cheap measurements are error prone and the error propagates. Localization techniques attempt to approximate the location of the nodes in the network by minimizing the cost of the estimated error as [22][16]. The cost function can be expressed as follows.

$$\tau = \sum_{j \epsilon n_i} |\hat{d}_{ij} - d_{ij}| \tag{1}$$

Here d_{ij} and $\hat{d_{ij}}$ denote the measured and estimated distances between nodes i and j respectively. n_i denotes set of neighbors of node i. Based on the prior observations and reports on the weakness of the aforementioned cost function we modify it to a constrained optimization problem through the penalty function method.

C. Penalty incorporated cost

Minimizing sum of the distance errors by equation (1) does not perform well when a subset of anchors form a straight line. Non-anchor nodes flip to the opposite side of the straight line and cause high localization error. These flips are common for all the localization algorithms approximating the location based on the function (1) and commonly known as flip ambiguity. In some cases not only one node flips but the whole neighborhood suffers from this phenomenon.

It is imperative in designing a localization algorithm such that the algorithm handles the flip ambiguity. A simple observation is, if the location of the node is flipped then it is likely the minimum hop count of the node from some other nodes will be altered. A constrained optimization algorithm by equating the hop counts can handle the issue to a large extent for a reasonably dense network.

For this specific case the optimization becomes minimizing equation (1) subject to $hm_i = he_i$. Here, hm_i and he_i are the measured and estimated hop distances of node i respectively.

Minimum hop counts can simply be implemented by Bellman-Ford's or Dijkstra's algorithm. Whenever there is a mismatch of hop count, the optimization algorithm can penalize the cost by monotonically increasing the penalty in successive rounds. But measuring this minimum hop distance must be implemented in every round of the optimization technique. Therefore the constrained hop count based optimization will become undesirable due to its processing cost.

The same result can be achieved by evaluating the neighbors for each node. If a specific node's estimated location reviles a missing neighbor from the measured neighbor list then a penalty is added as a cost. Similarly if there exists an additional node in the evaluated neighbor list that does not exist in the measured neighbor list a penalty is added. In this case the optimization problem becomes minimizing

$$\tau = \sum_{j \epsilon n_i} |\hat{d}_{ij} - d_{ij}| \tag{2}$$

$$\text{s.t. } d(N_i) < \Re;\ d(\bar{N}_i) > \Re$$

Here, N_i denotes the set of neighbor nodes and $\bar{N}_i$ denotes the set of non-neighbor nodes of node i in the network. And finally the transmit range of the nodes are denoted as $\Re$.

By using the penalty method we in turn change the cost function of a constrained optimization problem where with the new cost function the optimization problem becomes a general form of optimization (without any constraints). Different types of techniques such as augmented Lagrange method, penalty function method, quadratic programming etc. can be used to solve the inequality constrained optimization problem. We attempt to solve the problem by using penalty function method. Two different kinds of penalty methods are sequential and exact penalty transformation. Among the sequential methods there are exterior-point penalty method and barrier function method. Typically the barrier functions are inverse or logarithmic. We implement the logarithmic version of the barrier function method for our algorithm. We chose the barrier method because it preserve the feasibility at all times. Therefore the modified cost to be optimized can be expressed as $\tau = \sum_{j \epsilon n_i} |\hat{d}_{ij} - d_{ij}| + r_k * (-\sum_{j \epsilon n_i} \ln(-|\hat{d}_{ij} - \Re|) + r_k * (-\sum_{j \epsilon \bar{n}_i} \ln(-|\hat{d}_{ij} - \Re|)$. Here, r_k is the monotonically increasing penalty function based on rounds. Having no constraint the simple SPSA can handle the problem straightforward.

D. Localization algorithm

SPSA localization algorithm starts with randomly initializing locations in the x-y coordinates of all nodes in the network. It then randomly selects a single node of interest at that particular run and calls SPSA engine for approximating the node location based on the neighbor distances. Upon updating the location information receiving from SPSA algorithm, the localization algorithm in turn selects a second node. The procedure continues until the correction becomes less than a predefined threshold.

E. SPSA

Stochastic optimization is an appropriate solution when a closed form solution is hardly obtainable. SPSA is a stochastic optimization tool for the multivariate systems that converges iteratively to the optimal state. As other approximation techniques SPSA provides step by step technique from random initial guess to achieve final optimal value by improving the objective function in successive iterations. The input arguments in our case the measured distances are noisy information due to the random wireless propagation model.

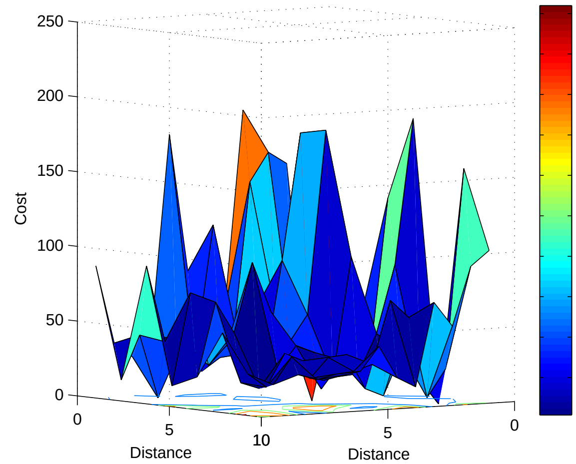
F. Selecting SPSA as a tool

Many optimization algorithms have assumption of deterministic settings requires information about the gradient vector associated to the cost function. Primarily wireless channel is random as a result the distances have similar randomness in the measurement error. Consequently the constituted cost function has largely irregular characteristic depicted in Fig. 1. Note that only two nodes with different distances have been taken to plot the figure because presenting more nodes is difficult because of increasing dimensions incorporated. Generally three or more nodes are required in the neighborhood to get a reasonable result for the location algorithms. Incorporating more nodes clearly will increase the complexity of the gradient. Therefore it is hard to obtain such gradient vector in our perspective though it could be an interesting future direction of our research. Such a problem can be handled by using large Monte Carlo simulations or by using recurrent neural networks. Gradient free approximation technique such as SPSA can provide similar convergence properties as gradient based approach therefore rather than digging deeper into the cost model for the functional relationship is avoided without sacrificing the error efficiency of the localization technique.

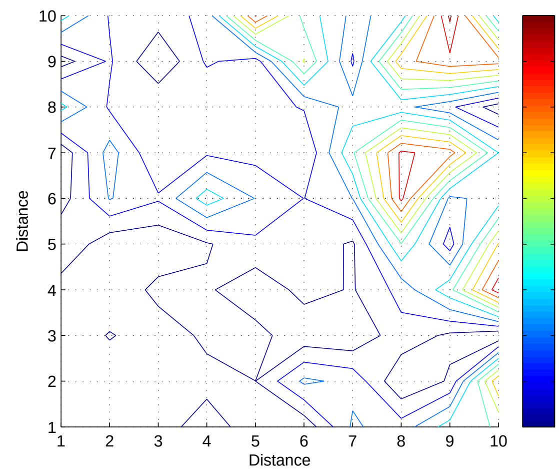
G. SPSA algorithm

Here, the primary problem is to minimize the differentiable cost function defined in equation (1). Let denote the cost function as $\aleph(\theta)$ where θ is a two dimensional vector contains x and y coordinates of the nodes to be estimated by the localization technique. It is trivial to find the cost for any given x and y. SPSA uses general recursive procedure to estimate θ by using the following equation.

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k \hat{g}_k(\hat{\theta}_k) \tag{3}$$



(a) Cost function



(b) Cost function contour

Fig. 1. Cost vs. neighbor distances

Where $\hat{g}_k(\hat{\theta}_k$ is the gradient estimation defined as $d\aleph/d\theta$ at k^{th} iteration. To evaluate the aforementioned gradient the loss function evaluation is performed as $\aleph(\theta \pm perturbation)$ as per the algorithm's name simultaneous perturbation designates. The elements of θ i.e., x and y randomly perturbed both in $+ve$ and $-ve$ directions to obtain two corresponding measurements. Therefore the gradient becomes

$$\hat{g}_{ki}(\hat{\theta}_k) = \frac{\aleph(\hat{\theta}_k + c_k \Delta_k) - \aleph(\hat{\theta}_k - c_k \Delta_k)}{2 c_k \Delta_{ki}} \tag{4}$$

where c_k denotes a small positive number and it decreases with successive iterations. Δ_k is known as random perturbation vector and can be expressed as equation as follows.

$$\Delta_k = [\Delta_{k1}^{-1} \Delta_{k2}^{-1} \ldots \Delta_{kn}^{-1}]' \tag{5}$$

This is an independent symmetrical vector distributed around 0 with finite inverse moment. Please note that a

SPSA localization technique
θ:(x, y)
Node level measurements for all node i
Use RSSI to measure neighbor distances
Update location server with neighbor distances
Algorithm level at central location
while $(Improvement > Threshold)$
{
Select a node randomly
Estimate the location of the selected node by $\digamma$ and $\aleph$
$\theta = \digamma(\theta)$
{
Tuning parameters: $n, p, a, c, \alpha, \beta, \gamma$
for $k = 1 : n$
{
$a_k = a/(k+\beta)^\alpha$
$c_k = c/k^\gamma$
$\delta = 2 * round(rand(p, 1)) - 1$
$\theta^+ = \theta + c_k * \delta$
$\theta^- = \theta - c_k * \delta$
$\tau^+ = \aleph(\theta^+)$
$\tau^- = \aleph(\theta^-)$
$\hat{g} = (\tau^+ - \tau^-)./(2 * c_k * \delta)$
$\theta = \theta - a_k * \hat{g}$
}
$\theta = \min(\theta, \theta_{max})$
$\theta = \max(\theta, \theta_{min})$
}
$\tau = \aleph(\theta)$
{
$r_k = 1/(\sigma^{(\varsigma - 1)})$
$\aleph = \sum_{j \epsilon n_i} |\hat{d}_{ij} - d_{ij}| +$
$r_k * (-\sum_{j \epsilon n_i} \ln(-|\hat{d}_{ij} - \Re|) +$
$r_k * (-\sum_{j \epsilon \bar{n}_i} \ln(-|\hat{d}_{ij} - \Re|)$
}
}

Fig. 2. Cross-entropy based localization algorithm

uniform or normal distribution does not satisfy the condition of choosing Δ_k. Rather a symmetric Bernoulli ± 1 distribution is a suitable candidate of Δ_k. Therefore Δ_{ki} is the $i^t h$ component of Δ_k vector and is a $\pm$ random variable.

Fig. 2 presents the proposed localization algorithm. The two core functions $\digamma$ and $\aleph$ in the figure represent the optimization technique and the cost function respectively. Here, $n, p, a, c, \alpha, \beta$ and γ are the tunable parameters. Where p is 2 in our case as the dimension of our estimation problem is 2. ς and σ are the current round and penalty factor of the algorithm. As we normalize the distance measures, θ_{max} and θ_{min} becomes 1 and 0 respectively. Other implementation that does not do any normalization can simply use their maximum and minimum values.

Therefore with the help of the penalty based cost function the proposed SPSA technique finds the estimated location of randomly chosen node. By iterative refinements all the location information of the non-anchor nodes is derived. Where the algorithmic inputs are the measured neighbor distances and initialized locations are mere random guesses.

IV. Simulation Results

We simulate our SPSA localization algorithm in Matlab and compare the results with those of SA. 100 nodes have been deployed in a 100m $\times$ 100m field. We compare the results for both random and fixed anchor deployments. And find the superiority of our proposal compared to SA. For fixed anchors we have $9, 16$ and 25 anchors equally distributed grid in the field where non-anchor nodes are always in random locations.

We model error in measurement by the following equation where d_{ij}^t and d_{ij}^m are the true and measured distances respectively. And n is the Gaussian disturbed random variables with mean 0.0 and variance 1.0. nf is the noise factor that regulates the magnitude of error. Note that the nf of the model is taken as 0.1 for all the experiments except for Fig. 9 and Fig. 10 where the nf varies from 0.2 to 1.0 with an increment of 0.1.

$$d_{ij}^m = d_{ij}^t * (1 + n * nf) \tag{6}$$

On the other hand, derived localization error is calculated as $error = (1/(N-A)) * (\sum_{i=A+1}^{N}((x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2)/R^2)$. Where, (x_i, y_i) and $(\hat{x}_i, \hat{y}_i)$ are the absolute and estimated locations of the node i. N and A are total number of nodes and total number of anchors in the network [16].

Fig. 3 shows convergence of the algorithm by depicting the error in rounds. Error exponentially decreases to a certain limit after such limit the decrease of error becomes almost negligible where running more for a better performance is not a cost effective solution. At this point we stop running optimization algorithms and capture the results. Fig. 4 shows the sensor field with 10 anchors where the locations are derived using SPSA and SA respectively. The distance measures are normalized where the radio range is set to 0.2.

Note that the algorithms were run ten times and root mean square error is taken as final result in the figures. Fig. 5 and Fig. 6 present error performance on the localization algorithms with grid anchor placements. The performance is measured for the transmission ranges $0.13 - 0.19$ for the cases $9, 16$ and 25 anchors. For all the cases SPSA performs better than the other algorithm.

Fig. 7 depicts error performance of the localizations with three different transmit rages $(0.18 - 0.2)$ for different number of random anchors $(5 - 50)$ in the field. For all the cases SPSA outperforms SA. Fig. 8 shows the error performance of the algorithms with different transmit ranges with 15 anchor nodes. For all the transmit ranges SPSA shows the superiority on the results.

Fig. 9a shows the localization error for the grid anchor deployments with respect to nf from $0.2 - 1$ where the radio range is set to 0.2. Fig. 9b shows the same result in 3-d space. SPSA only has high error with nine anchor deployments. While comparing to its counterpart it still exhibits error less

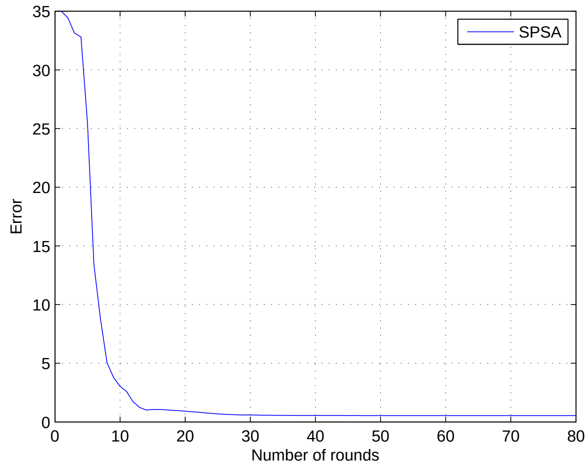


Fig. 3. Error in rounds

than half of SA error. For a larger set of anchors $(16/25)$ the error becomes negligible $(\leq 1\%)$ even with a very high noise factor. In this region SA is incomparably high.

Fig. 10a-b depicts the localization error with respect to the measurement error with randomly deployed anchor nodes in the field for various numbers of anchors. Where Fig. 10c shows the same result in 3-d space. Roughly similar performance exhibits in the random anchor deployment compare to the grid deployments as expected. Observing the slope of the curves we actually find the impact of nf on the error performance. Slopes in SPSA are much smaller than the slopes SA for the corresponding anchors. This trend reviles the fact SPSA is lot tolerant to nf compare to SA.

V. Conclusions and future works

To address the challenges in WSNs a novel localization algorithm based on constrained optimization is presented in this literature. The constrained optimization is achieved by modifying the cost function utilizing the barrier function based penalty function method. Finally the cost is optimized by SPSA algorithm. Simulation results show that a higher performance can be achieved by the proposed technique. The location information from the neighborhood may contain errors if the particular neighbor is a non-anchor node. Moreover the distance measurements are erroneous. Location coordinates derived based on the aforementioned errors adversely impact on the estimation. The errors are related to the node types (anchor/non-anchor), hop distances from the anchors, distances in meters etc. Therefore a trust model can be developed based on the underlying error posed by a neighbor node and the SPSA localization technique can be benefited by incorporating the model. Development of such model for SPSA localization is one of the aspects of our future directions. We also intend to observe the performance of the other penalty methods for our solution.

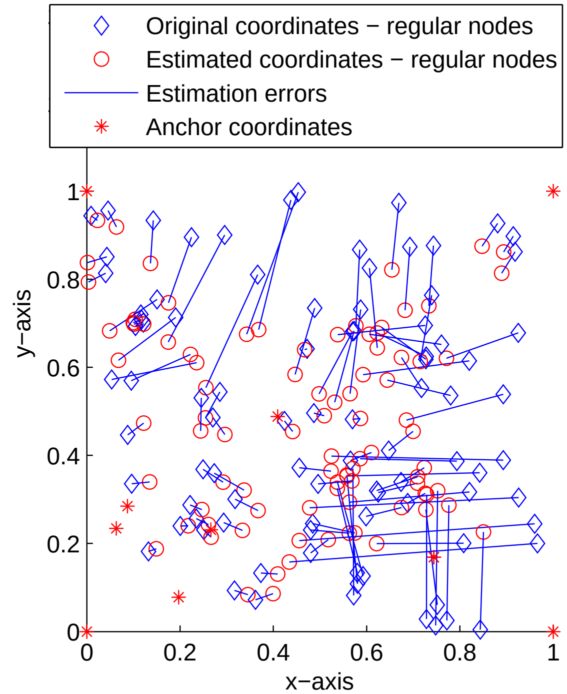
Acknowledgment

This research was sponsored by the Government of Abu Dhabi, United Arab Emirates through its funding of Masdar Institute's research project on "Monitoring and Optimization of Renewable Energy Generation using Wireless Sensor Data Analytics".

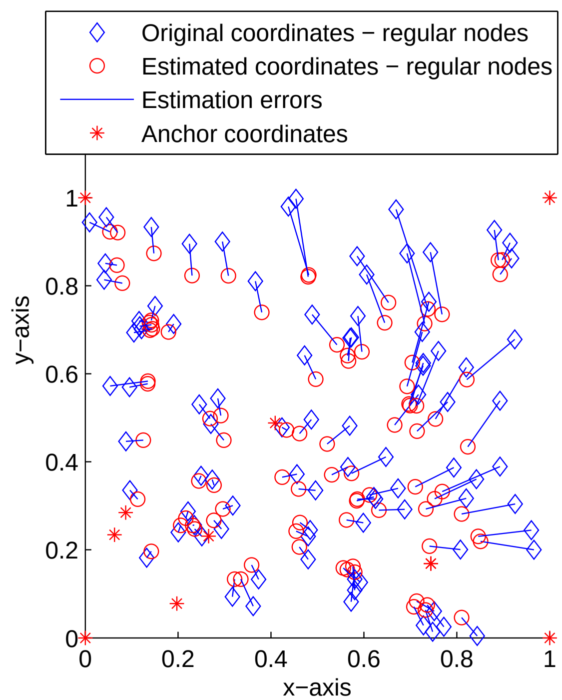
References

[1] J. Aitchison, and J. A. C. Brown, *The Lognormal Distribution*, Cambridge University Press, 1957.
[2] C. Alippi and G. Vanini, "A RSSI-based and calibrated centralized localization technique for wireless sensor networks," in *Proc. IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW)*, pp. 301-305, Pisa, Italy, March 2006.
[3] J. Bachrach, R. Nagpal, M. Salib, and H. Shrobe, "Experimental results for and theoritical analysis of a self-organizing a global coordinate system from ad hoc sensor networks," *Telecommunications System Journal*, vol. 26, no. 2-4, pp. 213-233, June 2004.
[4] J. Bachrach and C. Taylor, "Localization in sensor networks," *Handbook of Sensor Networks: Algorithms and Architectures*, 2005.
[5] R. Behnke and D. Timmermann, "AWCL: Adaptive weighted centroid localization as an efficient improvement of coarse grained localization," in *Proc. IEEE Positioning, Navigation and Communication*, pp. 243-250, Hannover, Germany, March 2008.
[6] J. Blumenthal, R. Grossmann, F. Golatowski, and D. Timmermann, "Weighted Centroid localization in Zigbee-based sensor networks," in *Proc. IEEE International Symposium on Intelligent Signal Processing*, Xiamen, China, Octobor 2007.
[7] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization for very small devices," *IEEE Personal Communications Magazine*, vol. 7 no. 5 pp. 28-34, October 2000.
[8] Y. Chan, W. Tsui, H. So, and P. Ching, "Time-of-arrival based localization under NLOS conditions," *IEEE Transactions on Vehicular Technology*, vol. 55 , no. 1, pp. 17-24, 2006.
[9] C. Chang and A. Sahai, "Estimation bounds for localization," in *Proc. IEEE Sensor and Ad Hoc Communications and Networks (SECON)*, Santa Clara, California, USA, October 2004.
[10] X. Cheng, A. Thaeler, G. Xue, and D. Chen, "TPS: a time-based positioning scheme for outdoor wireless sensor networks," in *Proc. IEEE INFOCOM*, pp. 2685-2696, Hong Kong, China, March 2004.
[11] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Section 24.3: "Dijkstra's algorithm," *Introduction to Algorithms (Second Ed.)*. MIT Press and McGraw-Hill. pp. 595601. 2001.
[12] T. Eren, D. Goldenburg, W. Whiteley, Y. Yang, A. Morse, B. D. O. Anderson, and P.N. Belhumeur, "Rigidity, computation, and randomization in network localisation," in *Proc. IEEE INFOCOM*, vol. 4, pp. 2673-2684, Hong Kong, China, March 2004.
[13] R. W. Floyd, "Algorithm 97: shortest path," *Communications of the ACM*, vol. 5, no. 6, pp. 345, 1962.
[14] L. Girod and D. Estrin, "Robust range estimation using acoustic and n multimodal sensing," in *Proc. IEEE International Conference on Intelligent Robots and Systems*, vol. 3, pp. 1312-1320, Hawaii, USA, November 2001.
[15] D. K. Goldenberg, A. Krishnamurthy , W. C. Maness, Y. Richard, Y. A. Young, A. S. Morse, A. Savvides, and B. D. O. Anderson, "Network localization in partially localizable networks," in *Proc. IEEE INFOCOM*, vol. 1, pp. 313-326, Miami, Florida, USA, March 2005.
[16] A. A. Kannan, G. Mao, and B. Vucetic, "Simulated annealing based wireless sensor network localization," *Journal of Computers*, vol. 1, no. 2, pp. 15-22, May 2006.
[17] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice Hall, pp. 344350, 1993.
[18] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," *in Proc. Second International Conference on Embedded Networked Sensor Systems (SenSys)*, pp. 50-61, Baltimore, Maryland, USA, November 2004.
[19] X. Nguyen and T. Rattentbury, *Localization Algorithms for Sensor Networks using RF Signal Strength*, University of California at Berkeley, Technical Report, May 2003.

[20] D. Niculescu and B. Nath, "DV based positioning in ad hoc networks," *Journal of Telecommunication Systems*, vol. 22, no. 1-4, pp. 267-280, 2003.
[21] A. Pal, "Localization algorithms in wireless sensor networks: current approaches and future challenges," *Network Protocols and Algorithms*, vol. 2, no. 1, pp. 45-78, 2010.
[22] N. Patwari, R. O'Dea, and Y. Wang, "Relative location in wireless networks," in *Proc. IEEE Vehicular Technology Conference (VTC)*, vol. 2, pp. 1149-1153, Rhodes, Greece, May 2001.
[23] R. Peng and M. L. Sichitiu, "Angle of arrival localization for wireless sensor networks," *in Proc. IEEE Sensor and Ad Hoc Communications and Networks (SECON)*, pp. 28-28, Reston, Virginia, USA, September 2006.
[24] N. Priyantha, H. Balakrishnan, E. Demaine, and S. Teller, *Anchor-free Distributed Localization in Sensor Networks*, MIT Laboratory for Computer Science, Technical Report TR-892, April 2003, Available: http://citeseer.ist.psu.edu/681068.html.
[25] A. Prorok, P. Tome, and A. Martinoli, "Accommodation of NLOS for ultra-wideband TDOA localization in single- and multi-robot systems," in *Proc. IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1-9, Portugal, September 2011.
[26] T. S. Rappapport, *Wireless Communications: Principles and Practice*, Prentice Hall: New Jersey, pp. 50-143, 1996.
[27] A. Savvides, H. Park, and M. Srivastava, "The bits and flops of the n-hop multilateration primitive for node localization problems," in *Proc. of the 1st ACM international Workshop on Wireless Sensor Networks and Applications (WSNA)*, pp. 112-121, Atlanta, Georgia, USA, September 2002.
[28] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, "Localization from mere connectivity," in *Proc. of ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pp. 201-212, Annapolis, Maryland, USA, June 2003.
[29] J. C. Spall, "An overview of the simultaneous perturbation method for efficient optimization," *Johns Hopkins APL Technical Digest*, pp. 482492, vol. 19, no. 4, 1998.
[30] S. Tian, X. Zhang, P. Liu, P. Sun, and X. Wang, "A RSSI-based DV-Hop algorithm for wireless sensor networks," in *Proc. IEEE International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 2555-2558, Shanghai, China, September 2007.
[31] J. Wang, P. Urriza, Y. Han, and D. Cabric, "Weighted centroid localization algorithm: Theoretical analysis and distributed implementation," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3403-3413, 2011.

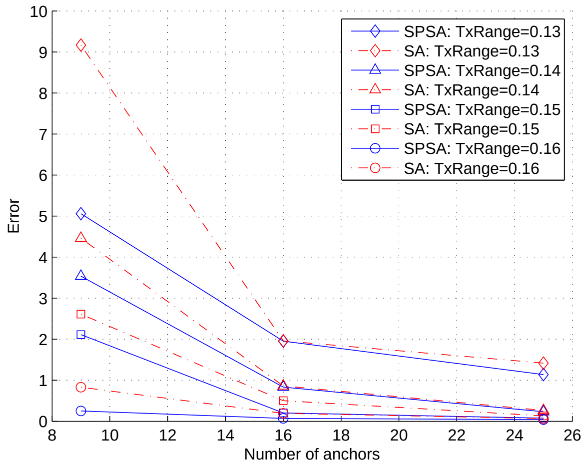


(a) Simulated annealing (SA)

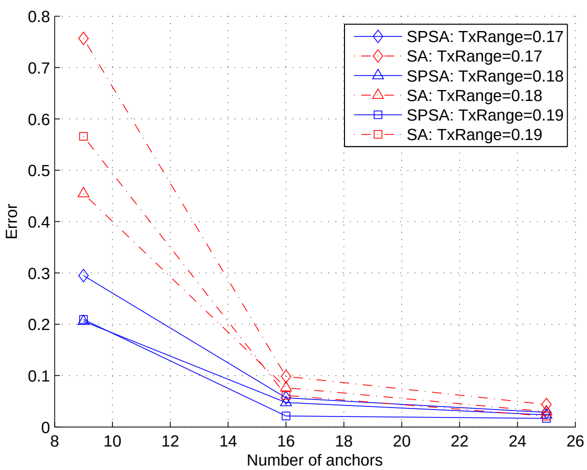


(b) Simultaneous perturbation stochastic approximation (SPSA)

Fig. 4. Original and estimated node locations in the network with 100 nodes among which 10 are anchors



(a) Tx range = 0.13-0.16



(b) Tx range = 0.17-0.19

Fig. 5. RMS error vs anchors (grid deployment)

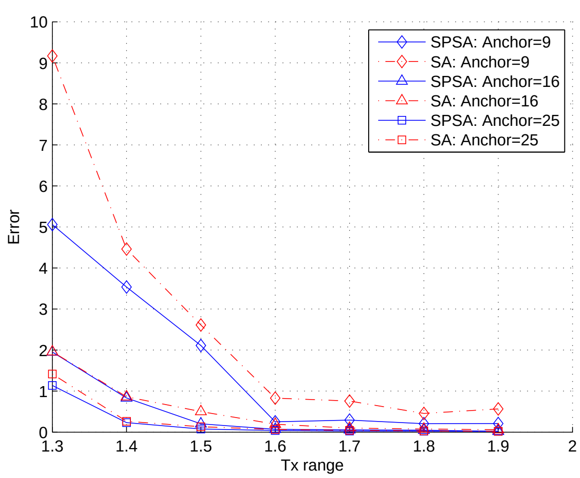
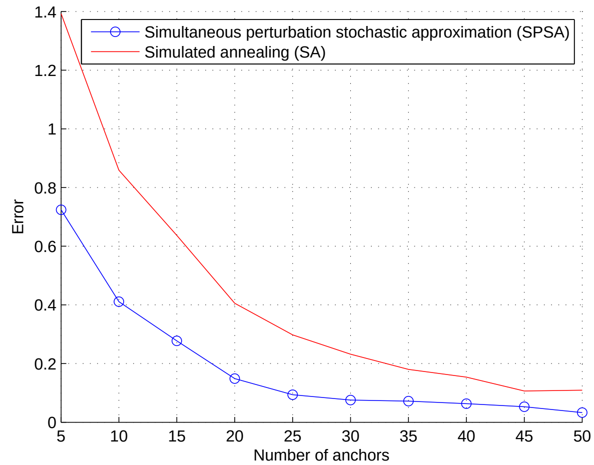
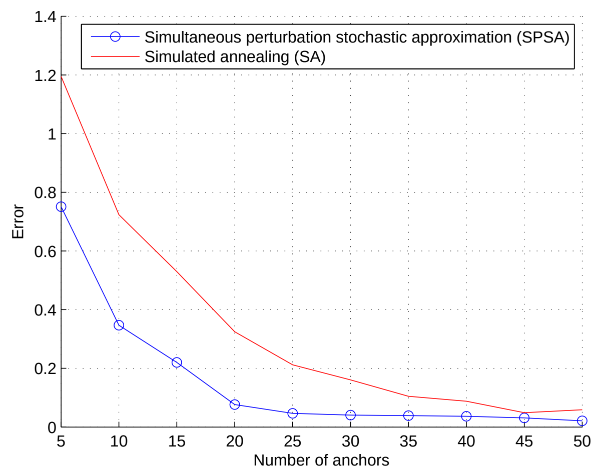


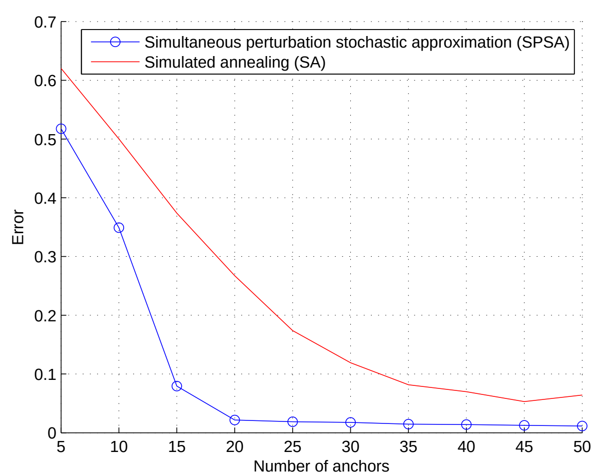
Fig. 6. RMS error vs transmit range (grid deployment)



(a) Tx range = 0.18



(b) Tx range = 0.19



(c) Tx range = 0.2

Fig. 7. RMS error vs anchors (random deployment)

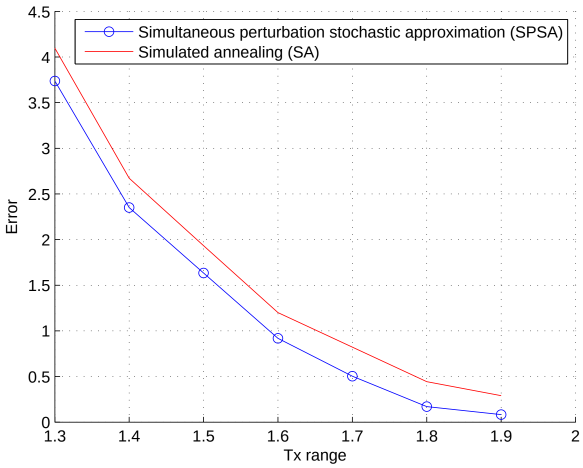
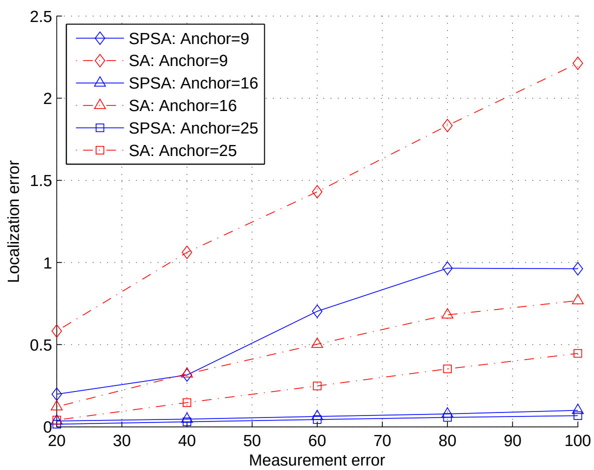
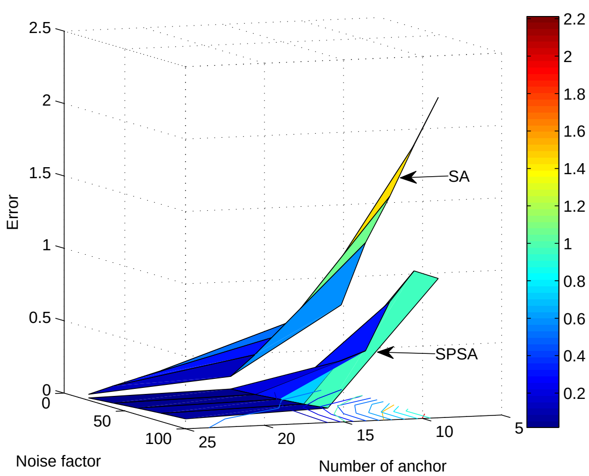


Fig. 8. RMS error vs transmit range (random deployment)

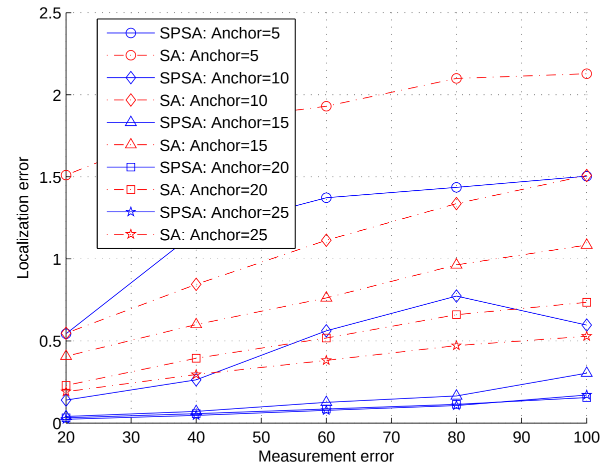


(a) Anchor: 9, 16 and 25

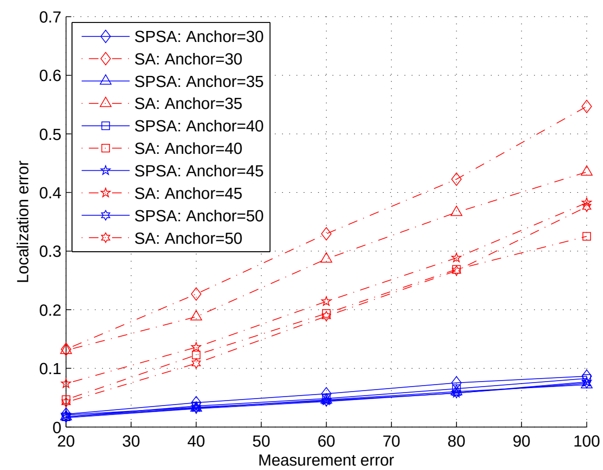


(b)

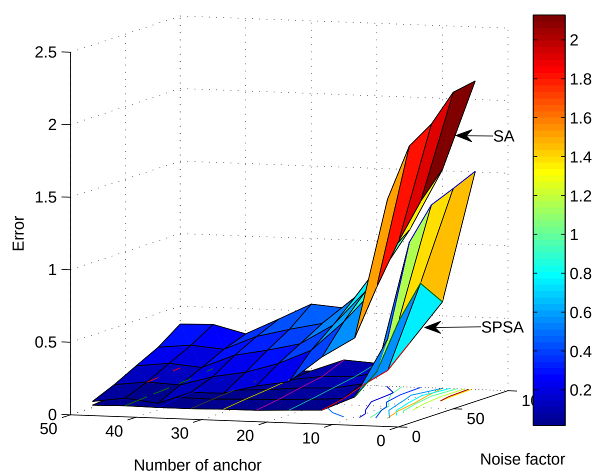
Fig. 9. Localization error vs measurement error (grid deployment)



(a) Anchor: 5, 10, 15, 20 and 25



(b) Anchor: 30, 35, 40, 45 and 50



(c)

Fig. 10. Localization error vs measurement error (random deployment)