Optimization of Wireless Locating in Complex Environments by Placement of Anchor Nodes With Evolutionary Algorithms

Tilman Leune, Thorsten Wehs, Manuel Janssen, Carsten Koch, Gerd von Cölln
Hochschule Emden/Leer
University of Applied Sciences, Germany
Department of Informatics and Electronics
Constantiaplatz 4 26723 Emden, Germany
{carsten.koch, gerd.von.coelln}@hs-emden-leer.de

Abstract

Lateration based real time locating systems (RTLS) require anchor nodes with known positions to calculate the position of a mobile target. The geometry of the anchor nodes constrains the accuracy of the locating. In complex environments such as ships or factory floors, obstacles in the line of sight between target and anchors decrease the precision significantly. This is caused by the influence of multipathing effects and shadowing on distance measurements. In this work, we propose a heuristic approach to find reasonable geometries for anchor nodes in complex environments. We achieve this by simulation of anchor geometries and the employment of evolutionary algorithms to search for optimizations.

1 Introduction

1.1 Motivation

Our long term goal is to realize a distributed Wireless Sensor Network (WSN) consisting of self locating mobile sensor nodes and fixed position anchor nodes used for spatial reference in hazardous environments [2].

In this work, the problem of optimal placement of reference anchors for communication and localization with lateration in heavily obstructed areas (e.g. on ships or on the shop floor) is explored. Since most lateration-based localization systems require line of sight (LOS) for range measurements, careful planning of anchor positions is required in complex environments with obstacles to avoid potential none line of sight (NLOS) connections.

Current real-world deployment of anchor nodes for WSNs is still heavily dependent on expert knowledge and trial-and-error to determine good anchor locations. The goal of this work is to develop an automated optimization software for a priori planning of anchor locations for a given complex environment, such as a ship or a factory.

1.2 Related work

In a number of previous work on WSN, the technological challenges of implementing wireless locating employing lateration [4] and signal distortion caused by obstacles [15] were explored. It was observed that the geometric layout of anchor nodes limits the maximum possible accuracy of locating. This is a well known property of lateration-based localization methods as described by Murphy et al. in 1995 [14]. For the 2D-case, the analytical proof of uniform perimeter placement of anchors being the optimal geometry for an obstacle-free environment has been presented by Ash and Moses in [1]. The introduction of systematic locating errors caused by non-optimal placement of anchor nodes and strategies to reduce these are object of current research [11]. Other research has been done on heuristic algorithms which employ DLS (Directed Local Search) strategies to optimize 3D placement of anchors in obstacle free areas in a way to minimize infrastructure cost while avoiding singularities and keeping appropriate positioning precision [8].

In general, while placement concepts and algorithms are well studied in two dimensions (2D), placement of anchor nodes in three dimensions has been found to be a NP-hard problem [7]. Assessment of the quality of a given 3D anchor node geometry can be performed by calculating the DOP-factor (dilution of precision), a unitless numeric value indicating the ratio of measurement noise to the estimated position deviation as proposed by Langley in 1999 [3] which has a very low computational complexity (compare Fig. 2).

The solution space of possible anchor node geometries for a target area with obstacles is infinite and finding an optimal solution is NP-Hard. A brute-force search is not feasible for the large search space. Using human intuition or expert knowledge to generate a good solution is difficult at best, and not suitable for larger environments with dozens or hundreds of anchor nodes. A given solution's quality can be assessed easily, though, so we propose a metaheuristic [5] approach employing evolutionary algorithms to find good and cost-effective solutions for anchor node placement in complex environments containing obstacles.

2 Hardware, simulation and genetics

2.1 Architecture and hardware basis of target WSN

This work is conducted for a specific wireless sensor networking system optimized for use as a RTLS (real time locating system) on ships, but the solution presented in this work is applicable for any anchor based RTLS dependent on LOS measurements.

The target network's architecture is modular, consisting of a small number of fixed position sensor nodes ('anchors') and mobile sensor nodes attached to the assets to be tracked. Each of the mobile nodes calculates its position from distance measurements to anchor nodes and transmits its current position, combined with other sensor data, to a central anchor component serving as a gateway to existing Ethernet infrastructure. This architecture is based on the proposals of Schulze and Wullner in 2006 [9].

The hardware of the sensor nodes consists of a custom built baseboard carrying a 32-bit microcontroller (LPCX-presso1769) and a commercially available UWB (Ultra-wideband) radio module [1] used for range measurements and communication [12].

2.2 Lateration

The unknown position $\mathbf{P_{lat}}$ (shown in Fig. 1) can be determined from the distances r_{0i} to at least four anchor nodes with known positions $\mathbf{P_i} = \{x_i, y_i, z_i\}$ where $i \in \{1, ..., n\}$ where $n \geq 4$. From the geometry of the known anchors and measured distances between them, the sum of distance squares is calculated using the Pythagoras theorem.

$$d_{ij}^2 = (x_i - x_1)^2 + (y_i - y_1)^2 + (z_i - z_1)^2 \quad (1)$$

Based on the distances d_{ij} between the four known anchors $\{\mathbf{P_1}, ..., \mathbf{P_4}\}$ and the position of $\mathbf{P_{lat}}$, a system of linear equations can be derived in the form of $A \cdot x = b$ where x contains the position $\mathbf{P_{lat}}$ to be determined.
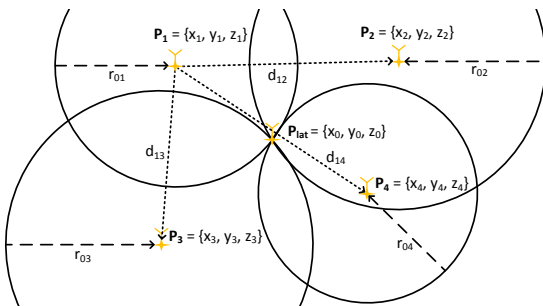
---

[1]timedomain RCM410

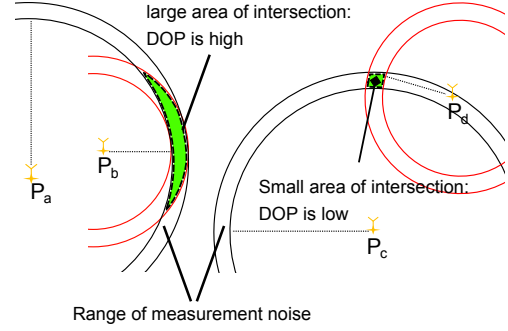Figure 1: Geometric dependencies of localization via lateration

Figure 2: Dilution of precision (DOP) caused by overlaid measurement noise leading to larger areas of solution

$$A = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ x_4 - x_1 & y_4 - y_1 & z_4 - z_1 \\ \vdots & \vdots & \vdots \\ x_n - x_1 & y_n - y_1 & z_n - z_1 \end{bmatrix} x = \begin{bmatrix} x_0 - x_1 \\ y_0 - y_1 \\ z_0 - z_1 \end{bmatrix}$$

$$b = \begin{bmatrix} r_{01}^2 - r_{02}^2 + d_{12}^2 \\ r_{01}^2 - r_{03}^2 + d_{13}^2 \\ r_{01}^2 - r_{04}^2 + d_{14}^2 \\ \vdots \\ r_{01}^2 - r_{0n}^2 + d_{1n}^2 \end{bmatrix} \cdot \frac{1}{2}$$

Since the distance measurements are estimations with inherent measurement noise, this equation system can only be approximated. In our approach, we use the linear least squares method, shown in equation 2 to do this [14].

$$x = (A^T \cdot A)^{-1} \cdot A^T \cdot b \quad (2)$$

The solution found in x is relative to the selected reference anchor $\mathbf{P_1}$. To transform x to the global coordinate system of the reference anchor, it has to be offset by $\mathbf{P_1}$

$$\mathbf{P_{lat}} = \mathbf{x} + \mathbf{P_1} \quad (3)$$

Two general types of systematic errors are present in this approach: First, there is the measurement noise σ, which is assumed to be normally-distributed zero-mean and inherent to our locating system's hardware. The second type of error is based on the DOP [3] caused by unfavorable anchor node geometries.

The minimization of the DOP-induced error however, especially when placing multiple anchor nodes in and around a complex environment, is studied in this work.

2.3 Numerical simulation of maximum possible accuracy

As a first evaluation, the mean numerical accuracy of different anchor geometries is calculated using a simulation engine without considering obstacles. For the 2D-case, the optimal geometry of anchors for maximized po-
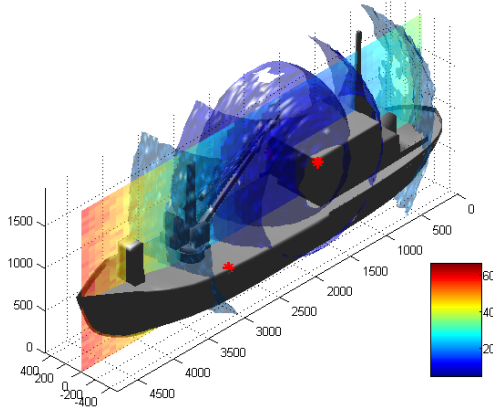
Figure 3: Simulation of RMSE with near real-world anchor geometry on a ship. The spheroid shells show surfaces of equal RMSE (all lengths in cm)

sition estimation accuracy has been proven to be a uniform placement around the perimeter of the WSN by Ash and Moses in 2008 [1]. The equivalent theoretical optimal anchor node geometry in three dimensions is placement on a tetrahedron (or on any regular polyhedron for overdetermined systems) around the area of interest [3].

The simulation is performed across a cuboid grid of $n_x \cdot n_y \cdot n_z$ grid points. In or around this grid, a number of fixed position anchor nodes is placed. For each grid point $\mathbf{P_{lat}}$ a lateration is calculated with zero mean Gaussian noise σ added to each true spatial distance measurement, with a standard deviation according to the noise characteristics of the target hardware used for the sensor network [13].

In Fig. 3, the results of such a simulation based on a σ of 3 cm are visualized.

Since the simulation approach does not account for obstacles in the target area, a way of selecting and evaluating anchor node geometries is required which does. In the following sections, a new method is proposed to select and evaluate anchor geometries based on the 3D topology of the target area given as triangular mesh.

3 Implementation

3.1 Genetic algorithms

Based on observations on biological evolution, the principle of gradual improvement by selection, recombination and mutation of individuals in a population can be used to find good solutions for high complexity decision problems. As depicted in figure 4, this approach of genetic algorithms (GA) is an iterative process.

Each individual in a population is represented by its genome which defines all its properties, explicit and implicit. Four mechanisms are needed for a basic implementation: a function calculating the *fitness* of an individual's genome, a method for *selection* of individuals as parents for the next generation, a method to *crossover* the genomes of these selected to create the next generation
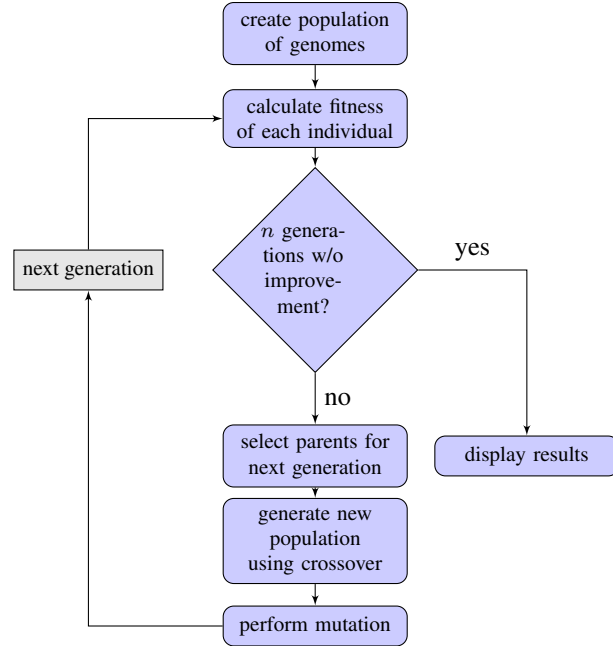

Figure 4: Flowchart of genetic algorithm

and finally a method to perform *mutation* of the new population's genomes.

Depending on population size and the cost of fitness calculation, the computational cost of genetic algorithms can be high. The calculation of fitness can be a very expensive operations for complex high dimensional problems since a single evaluation might include complex simulations running for hours or even days - for just one individual. This principle weakness of GA can be mitigated slightly with simultaneous multiprocessing (SMP) if the fitness of an individual has no interrelation to other individuals in a population, thus making the problem embarrassingly parallel.

3.2 Genome

For the problem of placing a minimal amount of anchor nodes in a way that every spot of a target area has LOS to enough nodes to enable localization, an individual solution is defined by its genome consisting of a set with a cardinality in the range of $[u, v]$ of anchor node positions $\mathbf{P_1}, \mathbf{P_2}, \ldots, \mathbf{P_n}$.

$$G = \{\mathbf{P_1}, \mathbf{P_2}, \ldots, \mathbf{P_n}\}$$

$$u \leq |G| \leq v$$

The set of all individuals is defined as the population T.

$$T = \{G_1, G_2, \ldots, G_n\}$$

3.3 Calculation of fitness

The fitness is calculated from a simulated visibility test in a gridded target area with $n_x \cdot n_y \cdot n_z = n_{xyz}$ grid points, obstructed by obstacles represented as triangles or triangular meshes. If a ray cast from a grid point to an

anchor intersects at least one obstacle, that anchor node is considered not visible from this grid point. With this simple algorithm, a list of visible anchor points for each grid point is compiled.

Using this list, the quality of the visible anchor geometry for each grid point is estimated by calculating the DOP metric which is defined as ratio of error of estimated location P_{lat} to distance measurement error d

$$DOP = \frac{\Delta P_{lat}}{\Delta d}$$

The total position estimation error σ can thus be calculated from the DOP determined for a geometry of anchors and the standard deviation of measurement noise σ_m.

$$\sigma = DOP \cdot \sigma_m$$

For classification of each grid point, a tolerance ϵ needs to be defined as the desired level of DOP at all grid points. Given this limit, the grid points are classified into three classes:

(a) *visible* - the grid point is surrounded by enough anchor nodes with LOS-connections to perform lateration with a DOP value less then a preselected tolerance ϵ

(b) *suboptimal* - the grid point has enough LOS connections to perform a lateration, but DOP is greater then the tolerance ϵ

(c) *invisible* - the grid point does not have enough LOS connections to anchor nodes to perform lateration

Fitness is calculated for each individual genome G_i of the population T. The parameters of the fitness function are the amount of invisible grid points n_i, the amount of suboptimal grid points n_s and the count of anchor nodes defined in the genome $|G_i|$. The parameter f_i is the ratio of invisible grid points to the total amount of grid points, f_s is the ratio of suboptimal grid points to total grid points and f_a is the ratio of employed anchors to maximum allowable anchor count. These parameters are scaled to values within the range of $[0, 1]$, allowing comparison of relative 'goodness' and 'badness' independent of the actual model parameters. The parameter f_a derived from the amount of anchors is offset to compensate the set minimum required number of anchors for locating.

$$f_i = 1 - \frac{n_i}{n_{xyz}}$$

$$f_s = 1 - \frac{n_s}{n_{xyz}}$$

$$f_a = \frac{v - (|G_i| - u)}{v - u}$$

The total fitness F is calculated as dot product of the parameter vector and a weighing vector of constant values w_i, w_s and w_a serving as tuning parameters of the model.

$$F(G_i) = \begin{pmatrix} f_i \\ f_s \\ f_a \end{pmatrix} \bullet \begin{pmatrix} w_i \\ w_s \\ w_a \end{pmatrix}$$

The results of the fitness function are normalized to $[0, 1]$ for comparability of results independent of model parameters such as n_{xy} or v.

$$\hat{F} = \frac{F(G_i)}{w_i + w_s + w_a}$$

The weight parameters allow to express preference of certain optimizations.

Additionally, the tolerance ϵ can be used to set the desired level of DOP across the target area. Over all generations, the individual with the highest fitness F is kept as the best known solution until either the end condition is met, or a fitter solution is found.

3.4 Selection method

During the selection phase, candidate individuals for recombination are drawn randomly from the population with a probability according to the relative fitness. Early experiments were performed with a simple uniform roulette selection [10], drawing individual genomes at random from the entire population weighted by their fitness. The probability $p_s(G_i)$ of a genome G_i being selected as a candidate for recombination is

$$p_s(G_i) = \frac{\hat{F}(G_i)}{\sum\limits_{j=1}^{|T|} \hat{F}(G_j)}$$

This shows that the probability of an individual genome being selected is directly proportional to its fitness. This proved not to be optimal: solutions converged to low local maxima only ($\hat{F}$ around 0.7) and ceased to improve over that plateau. This behavior is caused by the high selection pressure against weak individuals in the population. This is problematic since in order to find a better solution, the current local maximum has to be abandoned and an area in the search space with reduced fitness has to be traversed on the way to finding a better local or even global maximum. If the weak are culled to harshly, any individual leaving the current maximum has a high risk of being excluded from recombination and thus be eliminated.

Hence, a less elitist method of selection is required to keep diversity in the population. The method of tournament selection provides these properties [6, 10]. It is performed by repeatedly drawing an uniformly random sample of k individuals from the population and selecting the fittest individual from this subset. Using this scheme, individuals of high fitness have a high probability of being selected, but the weakest of the flock still have a chance of being selected at all.

3.5 Crossover and mutation

In the crossover phase, a new generation of individuals is created from the selected individuals of the parent generation. To allow predictions about the total run time of the algorithm, the population size is kept constant between generations. For each individual of the new generation, two uniformly random distinct parents are drawn

from the pool of potential parents. The genome of the child is constructed by uniformly randomly drawing half the genes (list of anchor node positions) of each parents genome, without replacement.

To avoid converging to the first local optimum found, each genome is mutated to add innovation to the population. Each gene is mutated separately, controlled by the parameters mutation rate σ_m and a probability value for creation and deletion of an anchor node preselected as θ_m.

At least one and up to three types of mutations are performed per genome:

> *translation* – each anchor node position $\mathbf{P_I}$ is displaced by a random vector $\vec{\Delta x}$ with length $\gamma \in N(0, \sigma_m)$
>
> *creation* – with a probability of θ_m, a new anchor node is spawned at a uniformly random position in the target area
>
> *deletion* – with a probability of θ_m, a uniformly random selected anchor node is removed from the genome

3.6 Break condition

Independent of the actual model parameters (grid size, range of anchor nodes count, tournament sizes, weights of fitness components), the overall fitness over time follows a pattern of diminishing returns: During the initial generations, the median fitness of the population converges to a plateau quickly. Once the plateau is reached, fitter solution are coming up less frequently and the gain of fitness between two consecutively found best solutions diminishes.

Better solutions may still be found, but the mean time between increases grows. A break condition is needed to allow the simulation to terminate eventually. Once the maximum amount of iterations to be performed without finding a better solution than the currently best known one is reached, the algorithm terminates and the current best known solution is proposed as a result.

4 Performance

The examples provided in this work are limited to two dimensions for clarity. The genetic algorithm is independent of the dimensionality of the grid and anchor nodes as it uses an abstract fitness-metric for it's decisions. This metric depends on another algorithm calculating the visibility of anchors for individual grid points which works in three and two dimensions according to the dimensions of grid points and anchor positions.

As shown in Fig. 5(d), the GA converges fast during the first iterations, but over time the improvement rate declines as the the fitness converges towards a maximum level. This level is the local optimum the GA can reach with the given parameter set. Improvements over this level are unlikely and will be small, so the break condition of

20-50 iterations without finding a better solution is advisable.

Profiling of a run of the GA shows that the major amount of runtime is spent evaluating the fitness of genomes. This is expected behavior since the other operations selection, crossover and mutation all iterate over the population once with a computational complexity of $O(n)$. In order to evaluate the fitness of an individual genome, however, a computationally expensive visibility test is performed. The visibility test is basically a ray-triangle intersection test for 3D-simulations. The computational cost increases linearly with the grid size n_{xyz} But, since the amount of grid points in a cuboid grows to the third power of the lengths of the cubes sides, so does the computational cost.

Let m be the number of obstacles, n be the number of anchor nodes and n_1, n_2, n_3 be the number of grid steps per dimension. This gives a computational complexity of

$$O(n_1 \cdot n_2 \cdot n_3 \cdot m \cdot n)$$

for a single visibility test. Since there are generally a lot more target grid points then there are anchor nodes, the influence of the amount of anchor nodes is insignificant for large n_i.

Since the visibility tests are performed once for each member of the population per iteration, the total complexity for one iteration of a population of size s = $|T|$ is

$$\lim_{n,m,s \to \infty} O(n^3 \cdot m \cdot s) = O(n^3)$$

In Fig. 6, a solution generated by the proposed algorithm is presented. The target areas have complete coverage and little to no areas of diluted precision. The amount of anchor nodes employed is below the maximum v. Convergence of fitness to a local optimum was observed in all simulation runs, independent of the model parameters (see Fig. 5(d)). The maximum level of fitness reachable with a given set of model parameters is less predictable. As long as the diversity of the population is kept large and deviation from the last found local optimum does not lead to assured elimination from the gene pool, a better solutions may be found eventually. Since there it is not known if the currently known best solution is a global or a local optimum (except if a individual reaches a fitness score of 1), the time available for simulation is the main constraint.

5 Conclusion

In this work, a heuristic approach was proposed to find solutions for anchor node placement in complex environments containing obstacles. A genetic algorithm capable of finding good solutions was presented based on a fitness function optimized for anchor placement in complex environments. It was also shown that the quality of solutions found by the GA depends on the parameter set for the modeled simulation.

The effect of co-evolution can be observed when performing multiple optimization runs with the same set of

(a) Generation 1, fitness = 0.517   (b) Generation 9, fitness = 0.893   (c) Generation 52, fitness = 0.958   (d) Spread of fitness over time
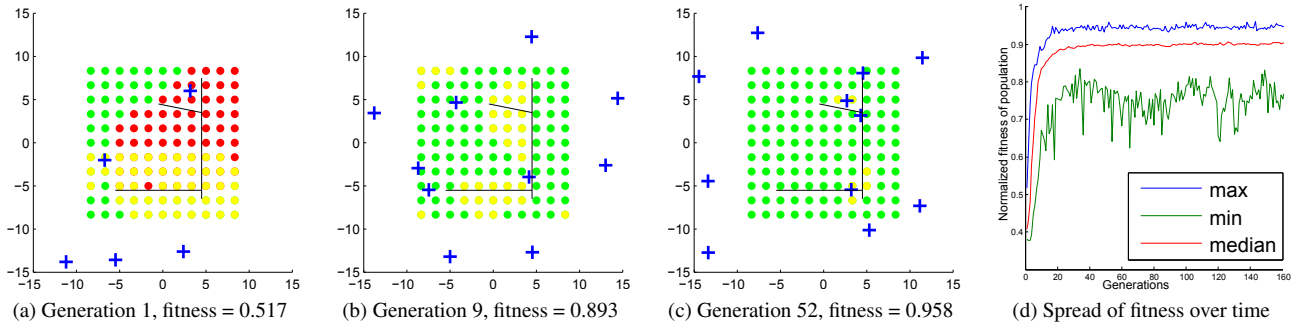
Figure 5: Example snapshots of a 2D solution showing the convergence of the genetic optimization over several generations. Red spots are non-locatable, yellow spots are locatable but with diluted precision and green spots are locatable with high precision.
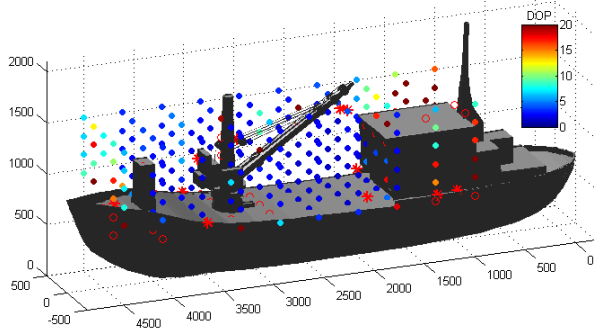

Figure 6: Optimized anchor node positions: 13 anchors cover work deck of a buoy tender with $DOP < 6$, locating error $\sigma < 6 \cdot \sigma_m$. Red hollow circles denote positions locating is not possible

obstacles and parameters. The 'double anchors' visible in Fig. 5(c) at the intersection points between the obstacle lines can be found in almost all solutions generated from the parameter set.

This effect hints at the possibility of stable design rules for good solutions which can be used as "'best practice'" without individual simulations to improve the proposed algorithm in the future. For a real-world implementation, some additional design rules must be implemented in the algorithm, such as constraining the anchor positions to feasible locations where they can actually by mounted on the existing infrastructure and more precise specification of the required precision per area covered.

References

[1] J. Ash and R. Moses. On optimal anchor node placement in sensor localization by optimization of subspace principal angles. In *Acoustics, Speech and Signal Processing, 2008*, pages 2289 –2292, 31 2008-april 4 2008.

[2] R. Droste, E. Böde, A. Hahn, T. Peikenkamp, and J. Lenk. Security analysis for offshore operations, 2011.

[3] R. B. Langley. Dilution of precision. *GPS World*, pages 52–59, May 1999.

[4] T. Leune, T. Wehs, M. Janssen, C. Koch, and G. von Cölln. Wireless locating and data communication in harsh industrial environments. In *Emerging Technologies and Factory Automation (ETFA), 2012 IEEE Conference on*, 2012.

[5] S. Luke. *Essentials of Metaheuristics*. Lulu, 2009.

[6] B. Miller and D. Goldberg. Genetic algorithms, tournament selection, and the effects of noise. *Complex Systems*, 9:193–212, 1995.

[7] S. Poduri, S. Pattem, B. Krishnamachari, and G. S. Sukhatme. Sensor network configuration and the curse of dimensionality. In *The Third IEEE Workshop on Embedded Networked Sensors*, 2006.

[8] J. O. Roa, A. R. Jiménez, F. Seco, J. C. Prieto, and J. Ealo. Optimal placement of sensors for trilateration: regular lattices vs meta-heuristic solutions. In *Proceedings of the 11th international conference on computer aided systems theory*, EUROCAST'07, pages 780–787, Berlin, Heidelberg, 2007. Springer-Verlag.

[9] L. Schulze and A. Wullner. The approach of automated guided vehicle systems. In *Service Operations and Logistics, and Informatics, 2006. SOLI '06. IEEE International Conference on*, pages 522 –527, june 2006.

[10] R. Sivaraj and T. Ravichandran. A review of selection methods in genetic algorithms. *International Journal of Engineering Science and Technology*, Vol.3, Issue 5:3792–3797, 2011.

[11] B. Tatham and T. Kunz. Anchor node placement for localization in wireless sensor networks. In *2011 IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 180 –187, oct. 2011.

[12] Time Domain. An overview of ultra wideband. *3rd Annual Winter Workshop for U.S. Army Vetronics Institute*, January 2004.

[13] Time Domain. *Data Sheet PulsON P400 RCM*, 2011.

[14] W. S. Murphy, Jr. and W. Hereman. Determination of position in three dimensions using trilateration and approximate distances. Technical report, Department of Mathematical and Computer Sciences, Colorado School of Mines, Golden, Colorado, October 1995.

[15] T. Wehs, M. Janssen, C. Koch, and G. von Cölln. System architecture for data communication and localization under harsh environmental conditions in maritime automation. In *IEEE 10th International Conference on Industrial Informatics*, 2012.