Decentralized Algorithms for Sensor Registration

Valentino Crespi
Thayer School of Engineering
Dartmouth College
Hanover NH 03755
Email: vcrespi@acm.org

George Cybenko
Thayer School of Engineering
Dartmouth College
Hanover NH 03755
Email: gvc@dartmouth.edu

***Abstract*— In this paper[1] we investigate a problem arising in decentralized registration of sensors. The application we consider involves a heterogeneous collection of sensors - some sensors have on-board Global Positioning System (GPS) capabilities while others do not. All sensors have wireless communications capability but the wireless communication has limited effective range. Sensors can communicate only with other sensors that are within a fixed distance of each other. Sensors with GPS capability are self-registering. Sensors without GPS capability are less expensive and smaller but they must compute estimates of their location using estimates of the distances between themselves and other sensors within their radio range. GPS-less sensors may be several radio hops away from GPS-capable sensors so registration must be inferred transitively. Our approach to solving this registration problem involves minimizing a global potential or penalty function by using only local information, determined by the radio range, available to each sensor. The algorithm we derive is a special case of a more general methodology we have developed called ”Emergence Engineering”.**

I. Introduction

Recent interest in sensor and surveillance systems based on unattended ground-based sensors (UGS) and unmanned autonomous vehicles (UAV's) has led to a proliferation of devices with different capabilities, sizes and costs. A typical goal is to deploy many small and inexpensive “easy-to-sacrifice” sensing devices with limited radio communication range that can form ad-hoc networks for communicating information back to processing stations and users. The sensing devices gather information about the surrounding environment (acoustic, seismic, infrared, temperature, humidity and so on) and then pass data through neighbors and ultimately to central processing or communications stations. Given the low envisioned cost of such technology, some number of sensor failures can be tolerated as long as the sensing requirements (like coverage of a certain area) and communications connectivity (for routing data back to users) are maintained. There are several examples of prototype and commercial sensors of this type [1].

In this work, we assume that sensor units consist of different components such as processors, memory, Global Positioning System (GPS) receivers, radio transmitter and various sensing modalities. Two specific types of sensors are considered - sensors with GPS capabilities and sensors without GPS capabilities. Sensors with GPS can of course self-register using the GPS signal while sensors without GPS must estimate their positions using information communicated with neighboring sensor units. The algorithm we present for the registration of all sensor nodes is iterative and decentralized. Our ultimate goal is to show that networks consisting of different types of sensors, expensive and cheap, can still be effectively used if the sensors collaborate on their registration. Our investigations have raised many interesting questions for future work. What are the tradeoffs between processing power and, for example, radio bandwidth and range? What density of GPS enabled devices is sufficient, with high probability, to register all sensor nodes within a given region, assuming some sort of random distribution of both types of sensors within that region.

Section II formulates the basic problem quantitatively. Section III describes our decentralized algorithmic approach, Section IV provides convergence results while Section V describes our experimental analysis. Finally Section VI discusses future work required in this area. We are currently conducting experiments and we have implemented this algorithm as a web browser accessible applet. [2]

II. Problem Definition

Let D_R be an open disk of radius R, denoting the planar region in which the sensors are deployed. (Note that the specific size and shape of the region is largely irrelevant to the algorithm and analysis we develop below.) A collection of n sensors $S = \{s_1, s_2, \ldots, s_n\}$ with equal radio range ρ are deployed in the region D_R. The following properties are defined for each sensor:

1) A “type” property $t : S \to \{G, N\}$; where $t(s) = G$ means the sensor s is GPS-enabled and $t(s) = N$ means it is “Not GPS-enabled”;
2) A “position” function $P : S \to D_R$ where $P(s) \in D_R$;
3) A “cost”, $c(s) > 0$, for each sensor.

Assuming that the m sensors labelled $s_1, s_2, ..., s_m$ are of type G (that is, have GPS capability), a fundamental problem is to position the $m \leq n$ GPS-enabled sensors from S and place them on a given disk D_R of radius R so that the following conditions hold:

- The totality of sensors provide sensing coverage of the whole disk D_R;
- The non-GPS sensors are able to infer algorithmically their absolute position by locally exchanging information

[1]Research supported in part by Defense Advanced Research Projects Agency Grant F30602-00-2-0585.

[2]See http://actcomm.dartmouth.edu/task/Demos/SensorsApplet/sensors.html

with the neighbors; that is, with the other sensors that fall within their radio communications range.
- The number of expensive, GPS-enabled sensors, m is minimized.

The quantification of the trade-offs should now be clear. The problem is to minimize the use of sophisticated, but costly, devices at the expenses of less expensive ones. This tradeoff requires having enough computational and communications power to compensate for less capable, non-GPS devices. In general, precise positioning of devices, either GPS or non-GPS enabled, is not possible because of how the sensors are deployed. They may be deployed to maximize coverage of subregions of high value or they may be literally dropped, as from an aircraft, and therefore end up with more or less random positions within the region.

Below, we describe a decentralized, iterative algorithm for non-GPS enabled sensors to self-register. We also present a preliminary experimental analysis of the required densities of GPS capable sensors to ensure, with high probability, self-registration of the whole sensor grid. The minimal such density, for a given probability of successful self-registration, would clearly be related to the problem of minimizing the cost of the overall sensor grid deployment.

III. Computing absolute positions

We approach the problem of inferring absolute positions of non-GPS equipped sensors deployed on a disk area D_R following a top-down methodology for the design of desired emergent behaviors in multiple agent-based systems (MAS) [2], [3], [4].

Assume that we are given a placement, P, of sensors from S. Without loss of generality, let $A = \{s_1, s_2, \ldots, s_k, s_{k+1}, \ldots, s_m\} \subseteq S$, where $t(s_i) = G$ for $1 \leq i \leq k$ and $t(s_i) = N$ for $k+1 \leq i \leq m$. This means that the first k sensors are GPS-enabled and the remaining ones are not. Let $P(s_i) = P_i \in D_R$ the true position of sensor s_i on D_R. By definition, the first k sensors are fully aware of their position $P_i \in D_R$ while the remaining $m - k$ sensors must determine their positions through a local computation and communication with their neighbors.

The basic idea behind the algorithm is as follows: each non-GPS sensor node starts with an initial random guess of its true position within D_R and then proceeds iteratively to successive refinements of that position estimate until the difference between two successive iterates becomes negligible. The goal of this process is for the position estimates to converge to the true values, $P_{k+1}, P_{k+2}, \ldots, P_m$.

Our technique is based on a global potential function to be minimized with a decentralized, possibly asynchronous, gradient descent method [5]. The effectiveness of the approach relies upon the fact that the gradient vector is locally computable. This is one of the key aspects of our top-down methodology in which the computation of a function requiring global resources can be computed or estimated using only local resources.

A. Artificial Potentials and Algorithms

Formally, let $X(t) = \{\mathbf{x}_i(t) \in \mathbf{R}^2 | 1 \leq i \leq m\}$ be the hypothesized positions of the sensors at time t. By definition, $\mathbf{x}_i(t) = P_i$ for all $t \geq 0$ if $1 \leq i \leq k$. That is, when s_i is a GPS-enabled sensor its position is known for all time. The problem is to define a potential function $V(X)$ such that $\nabla V(X)$ is locally computable and the following sequence

$$\begin{cases} \mathbf{x}_i(t+1) = \mathbf{x}_i(t) - \gamma_t \cdot \nabla_{\mathbf{x}_i} V(X(t)) \\ \mathbf{x}_i(0) = P_i^{(0)} \in D_R \end{cases}$$

converges to P_i as $t \to \infty$, for all $i > k$. That is,

$$\begin{cases} \mathbf{x}_i(t) = P_i & \text{for} \quad 1 \leq i \leq k \\ \mathbf{x}_i(t) \to P_i & \text{for} \quad k+1 \leq i \leq m \end{cases}$$

where γ_t is a suitable nonincreasing sequence of positive numbers (the "stepsize").

Let $d_{i,j} = \|P_i - P_j\|_2$ be the actual, true distance between sensor s_i and sensor s_j. By using radio signal strength, we are assuming that sensors can effectively estimate the distances between themselves and other sensors with which they can communicate, regardless of whether they are GPS enabled or not. That is, $d_{i,j}$ is known to both sensors s_i and s_j providing $d_{i,j} < \rho$, where recall that ρ is the effective communications range of the radio links. Note that while $d_{i,j} = \|P_i - P_j\|_2$ is known, the true positions of non-GPS sensor nodes, $P(s_j)$ for $j > k$, are not known but the estimates of $P(s_j)$, namely $\mathbf{x}_j$, are maintained on sensor s_j locally.

The potential function we consider is

$$V(X) \quad = \sum_{d_{i,j} < \rho} (\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 - d_{i,j}^2)^2 \ .$$

This potential function is composed of terms that are essentially the differences between the required inter-sensor distances, namely $d_{i,j}$ and the currently observed inter-sensor distances as determined by the positions estimates, $\mathbf{x}_i$. Consider the autonomous system described by the differential equation:

$$\dot{X} \quad = \quad -\nabla V(X) \ . \tag{1}$$

Then we can state the following facts:

- V can be expanded into a quartic polynomial and is continuously differentiable at least twice in $\mathbf{R}^{2m}$. This implies, among the others, that its partial derivatives $\nabla_{\mathbf{x}_i} V(X)$:

  $$\frac{\partial V}{\partial \mathbf{x}_i}(X) = 4 \cdot \sum_{j:\ d_{i,j} < \rho} (\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 - d_{i,j}^2) \cdot (\mathbf{x}_i - \mathbf{x}_j)$$

  are continuously differentiable and so ∇V is Lipschitz continuous on any bounded domain $B \subseteq \mathbf{R}^{2m}$ [6]:

  $$\exists K_B > 0\, \forall X, Y \in B\ \|\nabla V(X) - \nabla V(Y)\|_2 \leq K_B \|X - Y\|_2 \ .$$

- Function $V(X(t))$, where $X(t)$ is any trajectory of the system (1), satisfies in D_R^m: $\dot{V} = \nabla V \cdot X =$

$$= \sum_{i=1}^{m} \left[\frac{\partial V}{\partial x_{i,1}} \cdot \left(-\frac{\partial V}{\partial x_{i,1}} \right) + \frac{\partial V}{\partial x_{i,2}} \cdot \left(-\frac{\partial V}{\partial x_{i,2}} \right) \right]$$
$$= -\sum_{i=1}^{m} \left\| \frac{\partial V}{\partial \mathbf{x}_i} \right\|_2^2 \leq 0 \ .$$

- $V(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m) \geq 0$ in the open set D_R^m. Besides, $V(P_1, P_2, \ldots, P_m) = 0$. Thus V has an absolute minimum which is attained at $\mathbf{x}_i = P_i \ \forall i$.
- In general, the absolute minimum may be attained at many points. For example, any rototranslation of $(P_1, P_2, \ldots, P_m)$ is a solution to the equation $V(X) = 0$.

The assumption of having k GPS-enabled devices allows the introduction of an additional function:

$$\begin{aligned} U_k(X_k) &= U_k(\mathbf{x}_{k+1}, \mathbf{x}_{k+2}, \ldots, \mathbf{x}_m) \\ &= V(P_1, P_2, \ldots, P_k, \mathbf{x}_{k+1}, \ldots, \mathbf{x}_m). \end{aligned}$$

The following lemma is instrumental to the study of the stability of the dynamic system described by the equation

$$\dot{X}_k = -\nabla U_k(X_k)$$

using the Lyapunov's direct method [7].

Lemma 1: Assume we are given a set A of $m > 1$ deployed sensors as specified above. Let $G = (S, E)$ be an undirected graph defined as: $S = \{s_1, s_2, \ldots, s_m\}$ and $\{s_i, s_j\} \in E$ if and only if $d_{i,j} = \|P_i - P_j\| < \rho$. Moreover, assume that G is connected. Then the set $L = \{X_1 \in \mathbf{R}^{2(m-1)} \mid U_1(X_1) \leq c\}$ is bounded for every $c \in \mathbf{R}^+$.

Proof: Assume for simplicity that the origin of the coordinate system is centered at point P_1. Let $X_1 = (\mathbf{x}_2, \ldots, \mathbf{x}_m) \in L$ and, without loss of generality, let $m = \arg\max\{\|\mathbf{x}_j\| \mid j \geq 2\}$, so that $\mathbf{x}_m$ is the farthest point from P_1. The condition $U_1(X_1) \leq c$ implies that

$$\|\mathbf{x}_i - \mathbf{x}_j\| \leq \sqrt{d_{i,j}^2 + \sqrt{c}} = b_{i,j}, \ \forall i, j$$

Let $B = \max\{b_{i,j} \mid i, j\}$ a value that depends only on G and c. Then by the connectivity of G there must exist a path π in G labeled as $\pi = [s_{j_1}, s_{j_2}, \ldots, s_{j_l}]$, where $P(s_{j_1}) = P_1 = O$, that verifies the following inequalities

$$\|\mathbf{x}_m\|_2 \leq \sum_{i=1}^{l-1} \|\mathbf{x}_{j_i} - \mathbf{x}_{j_{i+1}}\| < l \cdot B \leq m \cdot B \ .$$

But this implies that L must be bounded.

■

This result can be easily generalized to k sensor positions $P_1, P_2, \ldots, P_k$. Of particular interest is the situation where equation

$$U_k(X_k) = 0 \quad (2)$$

is uniquely solvable because, in that case, U_k turns out to be a Lyapunov function for the autonomous system (1) and $(P_{k+1}, \ldots, P_m)$ is a point of stability in D_R^{m-k} [7]. Characterization of uniqueness does not seem very easy and is actually one of the challenges of this work.

It would be also very interesting to establish conditions under which equation 2 has only a finite number of solutions and consequently identify a subset $\Omega \subseteq D_R^{m-k}$ on which U_k is a Lyapunov function. We can use this property, wherever it holds, in order to prove the convergence of the asynchronous gradient descent algorithm.

IV. Convergence

We now focus on proving the convergence of our iterative decentralized gradient descent algorithm. We determine a bound for γ that ensures convergence to a stationary point (a zero of the gradient) although, sometimes, this might be a local minimum or a point of inflection as we verified experimentally (see IV-A). We proceed as follows.

- We first identify a bounded region that encloses the area on which our sensors will be placed. Assuming that each sensor is placed in a square of side L: $B = [-L/2, L/2] \times [-L/2, L/2]$ centered in the origin of the coordinate system, we consider the disk $D = D(\mathbf{0}, R)$, centered in the origin and having radius $R = \frac{\sqrt{2}}{2} L$ (the diagonal of the square).
- We establish a bound on the Lipschitz constant K of ∇U_k in D^{m-k}.
- We prove the validity of the *Descent* Lemma [8] relative to D^{m-k}.
- Finally we prove the convergence theorem of the method.

Let us start with the Lipschitz constant.

Theorem 2: Assume we are given a set A of $m > 1$ sensors deployed on a square $B = [-L/2, L/2] \times [-L/2, L/2]$ and be $D = D(\mathbf{0}, R)$ the disk that circumscribes it. Let $G = (S, E)$ be an undirected graph defined as: $S = \{s_1, s_2, \ldots, s_m\}$ and $\{s_i, s_j\} \in E$ if and only if $d_{i,j} = \|P_i - P_j\| < \rho$ and let d_M be the highest degree in G. Set $K_1 = 96 d_M \sqrt{2m} R^2$ and $K_2 = 96 d_M \sqrt{2} R^2$. Then $\forall X_k, Y_k \in D^{m-k}$

$$\begin{aligned} \|\nabla U_k(X_k) - \nabla U_k(Y_k)\|_2 &\leq K_1 \cdot \|X_k - Y_k\|_2 \\ \|\nabla_{\mathbf{x}_i} U_k(X_k) - \nabla_{\mathbf{x}_i} U_k(Y_k)\|_2 &\leq K_2 \cdot \|X_k - Y_k\|_2 \ . \end{aligned}$$

Proof: We know from basic calculus that for a function f of h variables that is in $C^{(1)}$ on a bounded convex set $Z \subseteq \mathbf{R}^h$ and for which $\forall i, \mathbf{x} \in Z \ |\nabla_{x_i} f(\mathbf{x})| < M$ it holds that

$$|f(\mathbf{x} + \mathbf{y}) - f(\mathbf{x})| \leq hM \cdot \|\mathbf{y}\|_2$$

Furthermore, if $F = (f_i)$ is a vector function of t components all $C^{(1)}$ on Z and whose partial derivatives are similarly bounded by M then

$$\begin{aligned} \|F(\mathbf{x} + \mathbf{y}) - F(\mathbf{x})\|_2 &= \sqrt{\sum_{i=1}^{t} |f_i(\mathbf{x} + \mathbf{y}) - f_i(\mathbf{x})|^2} \\ &\leq \sqrt{\sum_{i=1}^{t} (hM \cdot \|\mathbf{y}\|_2)^2} \\ &\leq hM\sqrt{t} \cdot \|\mathbf{y}\|_2 \ . \end{aligned}$$

So, to prove the theorem we need to consider the second partial derivatives[3] of V and find an upper bound M for them on D^m:

$$\frac{\partial^2 V(X)}{\partial x_{i,1}\partial x_{i',j}} = \begin{cases} -8(x_{i,1}-x_{i',1})(x_{i,j}-x_{i',j}) & \{s_i, s_{i'}\} \in E \\ -4(\|\mathbf{x}_i - \mathbf{x}_{i'}\|^2 - d_{i,i'}^2) & \\ & \\ 0 & \{s_i, s_{i'}\} \neq E \end{cases}$$

It is not hard to see that in the worst possible case we have

$$\left|\frac{\partial^2 U_k(X_k)}{\partial x_{i,1}\partial x_{i',j}}\right| \leq \left|\frac{\partial^2 V(X)}{\partial x_{i,1}\partial x_{i',j}}\right| \leq 48R^2 \ .$$

This corresponds to the case of estimates of very close points ($d_{i,i'} \approx 0$) that are very far from each other ($2R$). The claim now follows from the fact that in the first case $t = 2m$ and $h = 2d_M$, whereas in the second case $t = 2$ and $h = 2d_M$. ∎

Let us now discuss the Descent Lemma. Before doing this we need to anticipate the following result:

Lemma 3: Let K_1 and K_2 be the Lipschitz constants established before and $\gamma < \frac{1}{2K_2\sqrt{m-k}}$. Then

$$X_k \in D^{m-k} \implies -\gamma \nabla U_k(X_k) \in D^{m-k} \ .$$

Proof:

We show that $\forall i,\ k+1 \leq i \leq m\ \|-\gamma\nabla_{\mathbf{x}_i} U_k(X_k)\|_2 \leq R$. Let $X^* \in D^{m-k}$ be a point of global minimum for U_k. Then $\nabla U_k(X^*) = \mathbf{0}$. So

$$\begin{aligned} \|-\gamma\nabla_{\mathbf{x}_i} U_k(X_k)\|_2 &\leq \gamma \cdot \|\nabla_{\mathbf{x}_i} U_k(X^*) - \nabla_{\mathbf{x}_i} U_k(X_k)\|_2 \\ &\leq \gamma K_2 \|X^* - X_k\|_2 \\ &\leq \gamma K_2 \cdot 2R\sqrt{m-k} \\ &\leq R \ . \end{aligned}$$

∎

Let us now rephrase the Descent Lemma relative to a convex bounded set:

Lemma 4: Let $F : \mathbf{R}^n \to \mathbf{R}$ be a continuously differentiable function such that ∇F is Lipschitz continuous, on a convex bounded set Z, with constant K. Then $\forall x, y \in Z$:

$$F(x+y) \leq F(x) + y^{\mathrm{T}} \cdot \nabla F(x) + \frac{K}{2}\|y\|_2^2 \ .$$

Proof: Since Z is convex then $\forall t \in [0,1]\ x + ty \in Z$ and so the proof provided in [8] for the Descent Lemma on $\mathbf{R}^n$ immediately generalizes. ∎

Finally we can deliver the proof of convergence of the method that provides also a bound for the step value γ.

Theorem 5: Assume we are given a set A of $m > 1$ sensors deployed on a square $B = [-L/2, L/2] \times [-L/2, L/2]$ and be $D = D(\mathbf{0}, R)$ the disk that circumscribes it. Let $G = (S, E)$ be an undirected graph defined as: $S = \{s_1, s_2, \ldots, s_m\}$ and $\{s_i, s_j\} \in E$ if and only if $d_{i,j} = \|P_i - P_j\|_2 < \rho$ and let d_M be the maximum degree in G. Furthermore, let $K_2 = 96 d_M \sqrt{2} R^2$ and $K_1 = \sqrt{m} K_2$ be the Lipschitz constants for $\nabla_{\mathbf{x}_i} U_k$ and ∇U_k on D respectively.

Set $0 < \gamma < \frac{1}{2K_1}$. Then the sequence $\{X_k(t)\}$ generated by an algorithm of the form $X_k(t+1) = X_k(t) - \gamma \cdot \nabla U_k(X_k(t))$ satisfies

$$\lim_{t\to\infty} \nabla U_k(X_k(t)) = \mathbf{0} \ .$$

Proof: By definition we have $U_k(X_k(t+1)) = U_k(X_k(t) - \gamma \cdot \nabla U_k(X_k(t)))$ and by hypothesis $\gamma < \frac{1}{2K_1} = \frac{1}{2\sqrt{m}K_2} < \frac{1}{2K_2\sqrt{m-k}}$. So, by Lemma 3, $-\gamma \cdot \nabla U_k(X_k(t)) \in D^{m-k}$. But this allows us to apply Lemma 4 to obtain

$$\begin{aligned} U_k(X_k(t+1)) &\leq U_k(X_k(t)) + \\ &\quad -\gamma \nabla U_k(X_k(t))' \cdot \nabla U_k(X_k(t)) \\ &\quad + \frac{K_1}{2}\|\nabla U_k(X_k(t))\|_2^2 \\ &\leq U_k(X_k(t)) + \\ &\quad -\gamma\left(1 - \frac{K_1\gamma}{2}\right)\|\nabla U_k(X_k(t))\|_2^2. \end{aligned}$$

Our choice of γ ensures that the term $\beta = \gamma\left(1 - \frac{K_1\gamma}{2}\right) > 0$ and so now the proof proceeds as in [8] (Proposition 2.1): each value of $t \geq 0$ provides one such inequality. Summing for $\tau = 0, 1, \ldots, t$ we obtain the inequality

$$0 \leq U_k(X_k(t+1)) \leq U_k(X_k(0)) - \beta \sum_{\tau=0}^{t} \|\nabla U_k(X_k(\tau))\|_2^2 \ ,$$

true for all t. Then, in the limit

$$\sum_{\tau=0}^{t} \|\nabla U_k(X_k(\tau))\|_2^2 \leq \frac{1}{\beta} U_k(X_k(0)) < \infty \ ,$$

and so, necessarily, $\lim_{t\to\infty} \nabla U_k(X_k(t)) = \mathbf{0}$. ∎

Let us conclude this section with a result that establishes a sufficient condition for the uniqueness of the solution to the equation $U_k(X_k) = 0$.

Let us first proof the following lemma.

Lemma 6: Let $P_1, P_2, \ldots, P_s, P_{s+1}$ be a sequence of points on the plane with $s \geq 3$ and let $d_{i,j} = \|P_i - P_j\|_2$. Assume that there are at least 3 points among the first s that are not aligned. Then the equation

$$\sum_{i=1}^{s} (\|P_i - \mathbf{x}\|_2^2 - d_{i,s+1}^2)^2 = 0$$

has exactly one solution given by $\mathbf{x} = P_{s+1}$.

Proof: Clearly P_{s+1} satisfies the equation since $d_{i,s+1} = \|P_i - P_{s+1}\|_2$. The problem is to show that the solution is unique. We can observe that the equation is a sum of squares and so equivalent to the following system of quadratic equations: $\{\|P_i - \mathbf{x}\|_2^2 = d_{i,s+1}^2 \mid 1 \leq i \leq s\}$. Equation i: $\|P_i - \mathbf{x}\|_2^2 = d_{i,s+1}^2$ in the system is the equation of a circle centered at P_i and having radius $d_{i,s+1}$. Thus any solution will be given by the intersection of all the circles. Since there are at least three points not aligned then the intersection must

[3] In our notation $\mathbf{x}_i = (x_{i,1}, x_{i,2})$.

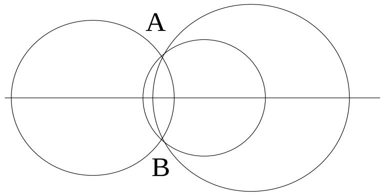

Fig. 1. Degeneracy due to alignment of the centers.

be unique. In fact, two circles intersect in at most two points, say A and B. If we pick a third point the only way for it to be at the same distance from A and B is to lie on the axis perpendicular to the line passing through A and B. But this means that the three centers would be perfectly aligned along that axis against the hypothesis. This situation of degeneracy is depicted in Fig. 1.

■

Let us now consider the following *Coloring* Algorithm.

1) Input: a set A of $m > 1$ deployed sensors as specified above. Let $G = (S, E)$ be an undirected graph defined as: $S = \{s_1, s_2, \ldots, s_m\}$ and $\{s_i, s_j\} \in E$ if and only if $d_{i,j} = \|P_i - P_j\|_2 < \rho$. Let the first k nodes in S be colored with black paint whereas let any other node in S be painted with a white paint.
2) If a white node is connected to at least 3 black nodes whose positions on the plane are not aligned then let it be blackened.
3) Repeat step 2 until no blackening is possible.

Theorem 7: Assume we are given a set A of $m > 1$ deployed sensors as specified before. Let $G = (S, E)$ be an undirected graph defined as: $S = \{s_1, s_2, \ldots, s_m\}$ and $\{s_i, s_j\} \in E$ if and only if $d_{i,j} = \|P_i - P_j\|_2 < \rho$. Furthermore, assume that $s_1, s_2, \ldots, s_k$, with $3 \leq k < m$, have been blackened. If the coloring algorithm blackens all the remaining nodes in S than the equation

$$U_k(X_k) = V(P_1, P_2, \ldots, P_k, \mathbf{x}_{k+1}, \ldots, \mathbf{x}_m) = 0$$

is uniquely solvable.

Proof: Observe that G is not necessarily connected. Let p be the number of iterations needed for the algorithm to complete and let A_i be the set of sensors that are blackened at the i^{th} iteration of the coloring algorithm. Then the following sequence of sets $B_i = \cup_{j \leq i} A_i$ verifies the following properties

- $B_0 = \{s_1, s_2, \ldots, s_k\}$;
- $B_i \subset B_{i+1}$;
- $B_p = S$.

Let us now define $G_i = (B_i, E_i)$ as the full subgraph of G having nodes B_i. By full we mean that whenever an edge $e = \{s_i, s_j\} \in E$ joins two nodes in B_i then it must be $e \in E_i$. Now, equation $V(P_1, P_2, \ldots, P_k, \mathbf{x}_{k+1}, \ldots, \mathbf{x}_m) = 0$ is equivalent to the following system H of quadratic equations:

$$\begin{cases} \|P_u - \mathbf{x}_v\|_2^2 = d_{u,v}^2 & \{s_u, s_v\} \in E \wedge 1 \leq u \leq k,\ v > k \\ \|\mathbf{x}_u - \mathbf{x}_v\|_2^2 = d_{u,v}^2 & \{s_u, s_v\} \in E \wedge u, v > k \end{cases}$$

Furthermore, each subgraph G_i determines a subsystem $H_i \subset H$ of equations

$$\begin{cases} \|P_u - \mathbf{x}_v\|_2^2 = d_{u,v}^2, \{s_u, s_v\} \in E_i \wedge s_u \in B_0 \wedge s_v \in B_i \backslash B_0 \\ \|\mathbf{x}_u - \mathbf{x}_v\|_2^2 = d_{u,v}^2, \{s_u, s_v\} \in E_i \wedge s_u, s_v \in B_i \backslash B_0 \ . \end{cases}$$

We shall prove, by induction on the number m of nodes of G, that, given any $i > k$, the only possible solution for variable $\mathbf{x}_i$ in system H is P_i.

- **Base:** $k = m$. In this case, clearly the system of equations H_0 is a trivial identity.
- **Induction Step:** Let us pick a node s_i in position P_i, with $i > k$. Suppose that s_i was blackened during iteration t of the algorithm. Then graph G_{t-1} must contain at least 3 neighbors of s_i whose positions are not aligned. Let those positions be $P_{j_1}, P_{j_2}, P_{j_3}$. Since $|B_{t-1}| < |B_t| \leq m$, we can apply the induction hypothesis to G_{t-1}. So all the equations in the corresponding system H_{t-1} involving variables $\mathbf{x}_{j_1}, \mathbf{x}_{j_2}, \mathbf{x}_{j_3}$ must be uniquely solvable and so $P_{j_1}, P_{j_2}, P_{j_3}$ are the only candidate solutions to equations in $H_t - H_{t-1}$ involving variables $\mathbf{x}_{j_i}$. But those include the three equations

$$\begin{aligned} \|P_{j_1} - \mathbf{x}_i\|_2^2 &= d_{j_1,i}^2 \\ \|P_{j_2} - \mathbf{x}_i\|_2^2 &= d_{j_2,i}^2 \\ \|P_{j_3} - \mathbf{x}_i\|_2^2 &= d_{j_3,i}^2 \end{aligned}$$

  that are uniquely solvable by Lemma 6. So P_i is the only candidate solution to all the equations in the big system H involving $\mathbf{x}_i$.

■

A. The Special Case: $n = 4$ and $k = 3$

Here we investigate the very special case of 3 GPS-enabled sensors plus a fourth device, within the range of all the previous three ones, that needs to self-register.

Assume that the GPS-enabled sensors have position $P_1 = \{x_1, y_1\}, P_2 = \{x_2, y_2\}, P_3 = \{x_3, y_3\}$, whereas the fourth device be placed at position $P_4 = \{x_4, y_4\}$. The potential function to be minimized will be:

$$\begin{aligned} V(x, y) &= \sum_{i=1}^{3} (\|P_i - \mathbf{x}\|_2^2 - \|P_i - P_4\|_2^2)^2 \\ &= ((x - x_1)^2 + (y - y_1)^2 - d_{1,4}^2)^2 + \\ & \quad ((x - x_2)^2 + (y - y_2)^2 - d_{2,4}^2)^2 + \\ & \quad ((x - x_3)^2 + (y - y_3)^2 - d_{3,4}^2)^2 \ . \end{aligned}$$

We verify numerically the existence of local minima of the scalar function V. First we run our Matlab simulation on random deployments of the sensors until the gradient descent process terminates on a stationary point other than P_4, say P_{4b}. Then we analyze the behavior of the second partial derivatives on P_{4b} to verify that it is a relative (local) minimum (and not a saddle point). This verification, of course, does not rule out the existence of saddle points. However, here we focus on local minima because those constitute the worst situation in a gradient descent approach.

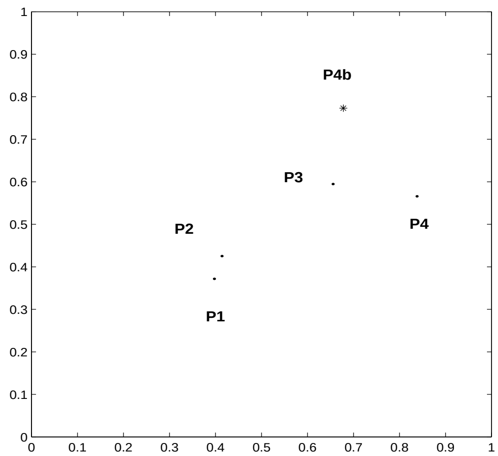

Fig. 2. A case of local minimum (P_{4b}) for $n = 4$, $k = 3$.

We apply standard techniques from Calculus based on expanding V, by Taylor's theorem, with a remainder of the third order, around a stationary point $\mathbf{x}$ and studying the sign of $Q(h, k) = V_{xx}(\mathbf{x})h^2 + 2V_{xy}(\mathbf{x})hk + V_{yy}(\mathbf{x})k^2$. Those techniques provide us with the following sufficient condition for a stationary point $\mathbf{x}$ to be a relative minimum for V [6]:

$$\begin{aligned} V_{xx}(\mathbf{x}) \cdot V_{yy}(\mathbf{x}) - V_{xy}(\mathbf{x})^2 &> 0 \\ V_{xx}(\mathbf{x}) &> 0 \end{aligned}$$

where $V_{xx} = \frac{\partial^2 V}{\partial x^2}$, $V_{yy} = \frac{\partial^2 V}{\partial y^2}$ and $V_{xy} = \frac{\partial^2 V}{\partial x \partial y}$.

Figure 2 reports one such a case. These results reveal that in general we should expect our algorithm to reach local minima and so we need to identify methods to precompute convenient initial values in order to avoid those traps in advance.

V. General Experiments

In this section we present the results of our experimental analysis. We search critical values of the percentage of GPS-enabled devices above which the algorithm converges with very high probability given a random deployment of sensors. We disseminate uniformly at random n sensors, having fixed communication range, on a square of side 1, then we run the gradient descent algorithm starting from random initial values. Figure 3 reports the estimated convergence probability $P(m)$ as m, the number of GPS-enabled devices, ranges from 3 to n. Intuitively the convergence process depends on the degree of connectivity of the graph G which is much higher for high values of the range to side ratio. For example, if the communication range is infinite and there are at least 3 GPS-enabled units we would expect the algorithm to converge with very high probability. So, as ρ tends to infinity, our expected curve will be a step function with discontinuity at point $m = 3$. As ρ tends to 0 instead the curve becomes more smooth. The conclusion is that there are no nontrivial critical points.

VI. Plan and Future Work

The following issues need to be explored in future work:

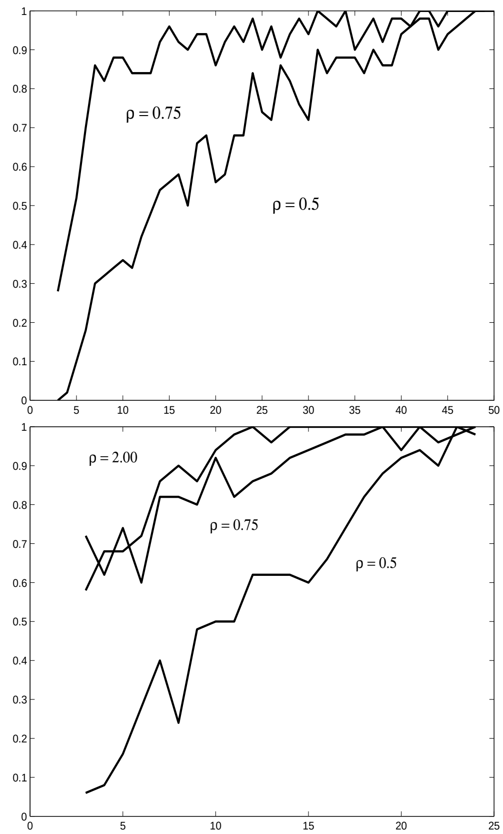

Fig. 3. Probability of converging to the right positions as the number of GPS units increases: (above) $n = 50$, $\rho \in \{0.5, 0.75\}$ (below) $n = 25$, $\rho \in \{0.5, 0.75, 2\}$.

1) Understand the structure of local minima better: do they occur in specific geometric structures?
2) Establish conditions on G that rule out the existence of local minima and, in general, of undesirable stationary points. In such cases, equation $U_k(X_k) = 0$ would be uniquely solvable and in addition $\dot{U}_k < 0$ on $\mathbf{R}^{2(m-k)}$.
3) Characterize unique solvability of equation $U_k(X_k) = 0$.

References

[1] M. G. Corr, "Geographic based ad-hoc routing for distributed sensor networks," Master of Science Thesis, Dept. of Computer Science, Dartmouth College, June 2001. [Online]. Available: http://agent.cs.dartmouth.edu/papers/corr:thesis.ps.Z
[2] G. Cybenko, "Agent-Based Systems Engineering," *Darpa Task Program Research Proposal. http://actcomm.thayer.dartmouth.edu/task/*, Oct 2000.
[3] V. Crespi, G. Cybenko, and D. Rus, "Decentralized Control and Agent–Based Systems in the Framework of the IRVS," *Darpa Task Program paper. http://actcomm.thayer.dartmouth.edu/task/*, Apr 2001.
[4] V. Crespi, G. Cybenko, D. Rus, and M. Santini, "Decentralized Control for Coordinated flow of Multiagent Systems," in *Proceedings of the 2002 World Congress on Computational Intelligence. Honolulu, Hawaii*, May 2002.
[5] D. Bertsekas and J. Tsitsiklis, "Gradient Convergence in Gradient Methods," *SIAM J. on Optimization*, vol. 10, pp. 627–642, 2000.
[6] R. Courant and F. John, *Calculus and Analysis*. John Wiley & Sons, 1974, vol. 2.
[7] J. L. Salle and S. Lefschetz, *Stability by Lyapunov's Direct Method*. Academic Press, 1961.
[8] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation*. Prentice Hall, 1989.