ITERATIVE LOCALISATION IN WIRELESS AD HOC SENSOR NETWORKS: ONE-DIMENSIONAL CASE

Aditya Karnik[*] *and Anurag Kumar*[†]

Department of Electrical Communication Engineering,
Indian Institute of Science, Bangalore, India 560 012

ABSTRACT

This paper discusses a simple method of localisation in sensor networks in which a sensor with unknown location is localised to a disk of radius equal to the transmission range centered at a beacon if the sensor under consideration can receive a transmission from the beacon. This is a reliable and extremely easy-to-implement technique since it assumes only a basic communication capability. The real advantage, however, is that once localised, a sensor aids the other sensors in localisation. This way by collaboration sensors can learn and improve their localisation regions iteratively. We analyse this iterative scheme and construct a distributed algorithm for utilising it in real sensor networks.

1. INTRODUCTION

Spatial or location information is of intrinsic interest in sensor networks; for example, it is essential to sensors for data combining and estimation. However, such information can neither be pre-configured in sensors owing to their ad hoc and possibly random deployment nor can it be centrally disseminated to sensors because of absence of a centralised co-ordinator. Further, because of cost and power constraints only a few sensors can be equipped with GPS receivers. Thus, it is imperative that sensors infer their locations autonomously using only a few sensors which have knowledge of their location either through mounted GPS receivers or a priori placement with preset co-ordinates.

This is not a new problem; locating objects in two (e.g., surface of the earth) or three dimensions (e.g., space) from the knowledge of locations of some distinguished nodes, called *beacons*, has been the central problem in navigation. Location of a node can be known from its distances and/or angles to beacons. For example, on the plane, if distances to at least three beacons are known from a node, then its co-ordinate can be fixed[1]. The distance measurements in this context are generally referred to as *ranging*. What distinguishes the localisation problem in sensor networks from the navigation problem is the following. Due to spatial expanse of a sensor network not every sensor will have the required number of beacons for ranging; to be cost effective, fewer beacons are desired. In addition, the traditional ranging methods based on received signal strength (RSSI), time of arrival (TOA), angle of arrival (AOA), time difference of arrival (TDOA), etc. ([3]) have several shortcomings from the point of view of the sensor networks. RSSI is usually very unpredictable since the received signal power is a complex function of the propagation environment. Hence, radios in sensors will need to be well calibrated otherwise sensors may exhibit significant variation in power to distance mapping. TOA using acoustic ranging will require an additional ultrasound source. TOA and RSSI are affected by measurement as well as non-line of sight errors. TDOA is not very practical for a distributed implementation. AOA sensing will require either an antenna array or several ultrasound receivers.

This motivates us to consider a particularly simple method of localisation which we call the *in-range* method (IR). The basic premise of IR is that a transmission at a given power can be decoded only up to a maximum distance, called its transmission range. IR then simply localises a node with unknown location to a disk of radius equal to the range centered at a beacon if the node under consideration can successfully decode a transmission from the beacon. Figure 1 shows an example of IR localisation. Let R denote the range and $D(x, r)$ a disk of radius r centered at x. In Figure 1, B_i is a beacon with location v_i for $i = 1, 2, 3$. The solid circles indicate disks $D(v_i, R),\ i = 1, 2, 3$. The dotted circles correspond to exact ranging. Thus, S is localised to $X = D(v_1, r_1) \cap D(v_2, r_2) \cap D(v_3, r_3)$, i.e., the region of intersection of three solid circles.

Though crude as compared to other ranging methods, IR is a reliable and easy to implement technique which assumes only a basic communication capability. The advantage of IR, however, is that once localised, a sensor can aid *the other sensors in localising and/or in reducing the localisation sets of some other previously localised sensors*.

[*] Research supported by a fellowship from the IBM India Research Lab.

[†] Research supported by a grant from the Indo-French Centre for the Promotion of Advanced Research (IFCPAR) (Project No. 2900-IT).

[1] Since no measurements are exact in practice, the location can only be estimated.

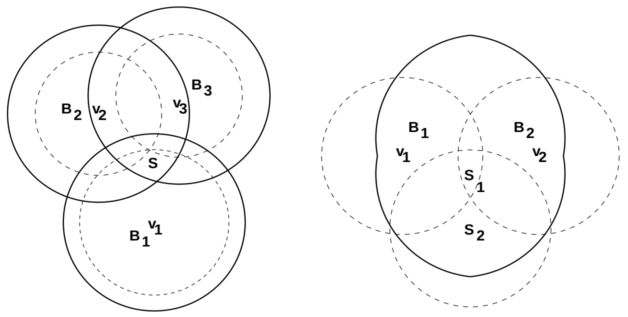

Fig. 1. Localisation with IR in the plane.

Figure 1 shows such a scenario. S_1 and S_2 are two sensors with unknown locations. S_1 is in the range of beacons B_1 and B_2; B_i located at v_i. Therefore, it gets localised to the region of intersection of $D(v_i, R),\ i = 1, 2$ shown by dotted circles with centers v_1 and v_2. S_2, though not in the range of either B_1 or B_2, is in the range of S_1; the dotted circle centered at S_1 is $D(v_{s_1}, R)$ where v_{s_1} denotes the location of S_1. Therefore, S_2 gets localised to the region bounded by the solid curve shown in Figure 1. *This way sensors can learn and improve their localisation sets iteratively.* The objective of this work is to investigate this iterative localisation process when the sensors and beacons are randomly placed and to construct a distributed algorithm for utilising this scheme in sensor networks. *In this paper we confine our discussion to one dimensional sensor networks.* Extensions to planar sensor networks will be discussed in an extended version of this paper.

This paper is organised as follows. Related work is reviewed in Section 2. In Section 3 we describe the general IR scheme and analyse it in the one dimensional case in Section 4; for brevity we omit the formal proofs which can found in [4]. Results are discussed in Section 5. A distributed algorithm based on this scheme is presented in Section 6. We conclude in Section 7.

2. RELATED WORK

The localisation scheme in [5] uses RSSI based distance estimates to beacons whereas in [1] it is based on TOA with acoustic ranging and multilateration. Both these methods have limitations discussed in Section 1. The basic premise of [2] is that communication hops between 2 sensors gives easy and reasonably accurate distance estimates at higher densities. Such distance estimates to many beacons are then used in a gradient descent algorithm at a sensor to minimise the location error. In its basic idea [6] is the closest to our work, i.e., a sensor with unknown location is localised to a "rectangle" centered at a beacon if the sensor can hear a "Hello" from it. However, the analysis is in a discrete setting and the localisation scheme is not iterative.

3. ITERATIVE LOCALISATION USING IR

Consider a randomly deployed sensor network in a geographical region $\mathcal{A}$; in this paper $\mathcal{A} \subset \mathbb{R}$. The sensors are indexed by $i \in \{1, 2, \ldots, N\}$. We say that a transmission can be "decoded" by a sensor when its signal to interference ratio (SIR) exceeds a given threshold β . The *transmission range* is then defined as the maximum distance at which a receiver can decode a transmitter in the absence of any co-channel interference. We denote the transmission range of sensors by R_0. The sensors within R_0 from i will be called its *neighbours*. The set of neighbours of i will be denoted by N_i and their number by n_i. By the location of sensor i we mean its co-ordinates and denote it compactly by v_i; in this paper v_i is just the x-coordinate of sensor i. A localisation set for a sensor i is a subset of the region of deployment. Let $X_i(0)$ denote the initial localisation set for $i = 1, 2, \ldots, N$. Recall that beacons are those sensors which know their location; the others are called *ordinary sensors* or simply sensors. Thus if i is a sensor then $X_i(0) = \mathcal{A}$ else $X_i(0) = \{v_i\}$. Recall that $D(v, r)$ denotes a disk of radius r centered at v; in one dimension disks are replaced by intervals. 0 denotes the origin. If G and H are two sets, $G + H$ denotes the set addition, i.e., $G + H = \{g + h | g \in G, h \in H\}$.

The following gives the iterative IR scheme. n is the iteration index. For $n \geq 0$ and $i = 1, 2, \ldots, N$,

$$Y_i(n+1) = \bigcap_{k \in N_i} (X_k(n) + D(0, R_0)) \qquad (1)$$

$$X_i(n+1) = X_i(n) \cap Y_i(n+1) \qquad (2)$$

$X_k(n)$ denotes the localisation set of node k in iteration n. Therefore, if i is in the range of k, i is certainly in the region $X_k(n) + D(0, R_0)$. Since this property holds for each neighbour of i, i is localised to $\bigcap_{k \in N_i} (X_k(n) + D(0, R_0))$. Thus, it follows that the localisation set of i in the $(n+1)^{th}$ iteration is the intersection of its localisation set in the n^{th} iteration and $\bigcap_{k \in N_i} (X_k(n) + D(0, R_0))$.

Let $\mathcal{L}(X)$ denote a measure of set X; in one dimension it is the length of X. Define, $\chi_i(n) := \mathcal{L}(X_i(n))$. We call $\chi_i(n)$ the *localisation error* of sensor i in iteration n. Let $\underline{\chi}(n) := (\chi_1(n), \chi_2(n), \ldots, \chi_N(n))$ and consider the vector valued process $\{\underline{\chi}(n),\ n \geq 0\}$ which we call the localisation process. Note from (2) that for each i, $\chi_i(n)$ is non-increasing with n. Let A denote $\mathcal{L}(\mathcal{A})$. We say that sensor i is localised by (i.e., at or before) iteration n if $\chi_i(n) < A$. Then the performance measures which are of interest are $\bar{\chi}(n) := \frac{1}{N}\sum_{i=1}^{N} \chi_i(n)$ and $\nu(n) := \frac{1}{N}\sum_{i=1}^{N} 1_{\{\chi_i(n) < A\}}$ where, $1_{\{.\}}$ denotes the indicator function. Thus, by definition $\bar{\chi}(n)$ is the average localisation error in the network at the end of n iterations and $\nu(n)$ is the fraction of nodes localised by iteration n.

4. ONE DIMENSIONAL LOCALISATION PROCESS

We assume N to be very large and model the random dispersion of sensors on the real line as a one dimensional Poisson point process Ψ of intensity λ; Poisson points indicate locations of sensing nodes. A fraction p of the nodes are beacons. In the point process model we assume that a node is a beacon with probability p independent of anything else. It follows that locations of beacons and sensors form independent Poisson processes of intensity λp and $\lambda(1-p)$) respectively.

Recall that v_j denotes the x co-ordinate of j. Thus, disk $D(v_j, R_0)$ extends from $v_j - R_0$ to $v_j + R_0$. $\chi_j(n)$ is the length of $X_j(n)$. The length of the interval lying to the right of j is denoted by $\Delta_j^r(n)$ while the length of the interval lying to the left of j is denoted by $\Delta_j^l(n)$; hence, $\chi_j(n) = \Delta_j^r(n) + \Delta_j^l(n)$. If j is a beacon, $\chi_j(n) = \Delta_j^r(n) = \Delta_j^l(n) = 0$, $n \geq 0$. If j is an ordinary sensor, $\Delta_j^r(0) = \Delta_j^l(0) = \frac{A}{2}$. In the point process model, A should be interpreted as the initial uncertainty preset in each sensor. The evolution (2) in this setting is as follows. Recall that N_j denotes the set of neighbours of j. For $n \geq 1$ and $j = 1, 2, \ldots, N$,

$$\begin{aligned}
u_j^r(n) &= \arg\min_{k \in N_j} (v_k + \Delta_k^r(n-1)) && (3)\\
\Delta_j^r(n) &= \min(\Delta_j^r(n-1), \\
&\quad v_{u_j^r(n)} + \Delta_{u_j^r(n)}^r(n-1) + R_0 - v_j) && (4)\\
u_j^l(n) &= \arg\max_{k \in N_j} (v_k - \Delta_k^l(n-1)) \\
\Delta_j^l(n) &= \max(\Delta_j^l(n-1), \\
&\quad v_j - v_{u_j^l(n)} + \Delta_{u_j^l(n)}^l(n-1) + R_0) \\
\chi_j(n) &= \Delta_j^r(n) + \Delta_j^l(n)
\end{aligned}$$

If N_j is empty, then by convention minimum over an empty set is taken to be ∞ and we define the location of $u_j^r(n+1)$ to be ∞. Similarly for $u_j^l(n+1)$.

To understand the iterative process given by (3), let us first consider $n = 1$. Assume that j is a sensor. Since $\Delta_k^r(0) = 0$ if k is a beacon and $\frac{A}{2}$ otherwise, it follows that, $X_j(1)$ is decided only by the beacons in its range. Further, $X_j(1)$ will be determined by the leftmost and the rightmost beacon in the range of j; the leftmost beacon will determine $\Delta_j^r(1)$ and the rightmost beacon will determine $\Delta_j^l(1)$. Step (3) locates the leftmost beacon; it is denoted by $u_j^r(1)$. Then it is easy to see that $\Delta_j^r(1) = v_{u_j^r(1)} + R_0 - v_j$ since R_0 is the range and v_j is j's location. Similarly $u_j^l(1)$ denotes the rightmost beacon so that $\Delta_j^l(1) = v_j - v_{u_j^l(1)} + R_0$.

Now for $n \geq 2$, consider an example shown in Figure 2. Assume that j has only one neighbour denoted by s_1 with location v_1. Suppose that s_1 is a sensor. Further, $n-1$ iterations are over; j has been localised to $[v_j - \Delta_j^l(n-1), v_j + \Delta_j^r(n-1)]$ and s_1 to $[v_1 - \Delta_1^l(n-1), v_1 + \Delta_1^r(n-1)]$. Since j lies in the transmission range of s_1 by virtue of this single constraint, j must lies within $[v_1 - \Delta_1^l(n-1) - R_0, v_1 + \Delta_1^r(n-1) + R_0]$ shown by the small square brackets. The length of the "right" side of this interval is $v_1 + \Delta_1^r(n-1) + R_0 - v_j$. It follows that $\Delta_j^r(n)$ will be the minimum of $\Delta_j^r(n-1)$ and $v_1 + \Delta_1^r(n-1) + R_0 - v_j$. Similar analysis applies to $\Delta_j^l(n)$. Equations (3) and (4) simply extend this logic to a general case.

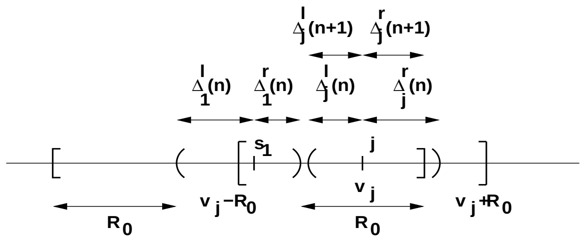

Fig. 2. An example of the propagation of $\Delta_j^r(n)$ and $\Delta_j^l(n)$.

Though (3) and (4) are much simplified compared to (2), $\underline{\chi}(n)$ is still not amenable to analysis. We now work with a typical point of the Poisson process, called the tagged node (denoted by o) and study the "marginal" process of $\underline{\chi}(n)$, i.e., $\{\chi_o(n),\ n \geq 0\}$, the sequence of localisation errors of the tagged node. The performance measures discussed in Section 3 can be obtained at the tagged node as, $\bar{\chi}(n) = E\chi_o(n)$, $\nu(n) = P(\chi_o(n) < A)$.

4.1. The Marginal Process, $\{\chi_o(n), n \geq 0\}$

If o is a sensor then $X_o(0) = \mathcal{A}$ else $X_o(0) = \{0\}$. The earlier discussion has shown that if o is a sensor $X_o(1)$ is decided only by the beacons in the range of o. It is thus possible to explicitly characterise the distribution of $\chi_o(1)$ ([4]). However, for further analysis we will work with a simpler process, $\{\Delta_o^r(n), n \geq 0\}$. Note that, $\Delta_o^r(1)$ and $\Delta_o^l(1)$ are identically distributed though not independent. By symmetry, this property holds for $n \geq 2$.

Proposition 4.1 *If o is a sensor, the probability distribution of $\Delta_o^r(1)$ is,*

$$P(\Delta_o^r(1) \leq y) = \begin{cases} 1 & y > \frac{A}{2} \\ 1 - e^{-\lambda p 2 R_0} & 2R_0 < y < \frac{A}{2} \\ 1 - e^{-\lambda p y} & 0 < y \leq 2R_0 \end{cases} \quad (5)$$

with probability mass at $A/2$: $P(\Delta_o^r(1) = A/2) = e^{-\lambda p 2 R_0}$.

Corollary 4.1

$$\frac{E\Delta_o^r(1)}{(1-p)} = \left(\frac{1 - e^{-\lambda p 2R_0} - \lambda p 2 R_0 e^{-\lambda p 2 R_0}}{\lambda p} + \frac{A}{2} e^{-\lambda p 2 R_0} \right)$$

Observe from (3) that, for a given $\Delta_o^r(1)$, $\Delta_o^r(n),\ n \geq 2$ is determined by the (ordinary) sensors in the range of o. Let N_o^s denote the set of sensors in the range of o and n_o^s their number. Consider now the iteration (3) applied to o for $n \geq 2$. $u_o^r(n) := \arg\min_{k \in N_o^s} (v_k + \Delta_k^r(n-1))$ and

$$\Delta_o^r(n) = \min(\Delta_o^r(n-1), v_{u_o^r(n)} + \Delta_{u_o^r(n)}^r(n-1) + R_0) \quad (6)$$

Recall that if N_o^s is empty, by convention minimum over N_o^s in (6) is infinite and $\Delta_o^r(n) = \Delta_o^r(n-1)$. Now a direct analysis of (6) amounts to analysing $\{\underline{\chi}(n),\ n \geq 0\}$ since to find the probability distribution of $\Delta_o^r(n)$, we need the joint distribution of $\Delta_k^r(n-1),\ k \in N_o^s$. However, an asymptotically exact approximation for the sequence $\{\chi_o(n),\ n \geq 2\}$ can be obtained as follows.

Since o is a typical point of Poisson process, n_o^s is Poisson distributed with mean $\lambda(1-p)2R_0$. We denote by o_k the k^{th} "sensor-neighbour" of o. For a given n_o^s, v_{o_k} are independent uniformly distributed random variables in $[-R_0, R_0]$. We now index these neighbours by $i \in \{1, \ldots, n_o^s\}$ based on the order statistics of v_{o_k}, i.e., the sensor corresponding i^{th} smallest value of v_{o_k}'s is indexed i. Thus 1 ($\arg\min_{1 \leq k \leq n_o^s} v_{o_k}$) is the "leftmost" neighbour and the rest in the increasing order towards right. Location of i is denoted by v_i; $f_{v_1}(x|n_o^s = m) = \frac{m}{2R_0}\left(1 - \frac{x+R}{2R_0}\right)^{m-1}$. Now consider a sequence $\{\hat{\Delta}_o^r(n),\ n \geq 0\}$ such that $\hat{\Delta}_o^r(1) = \Delta_o^r(1)$ and for $n \geq 2$,

$$\hat{\Delta}_o^r(n) = \min(\hat{\Delta}_o^r(1), v_1 + \hat{\Delta}_o^r(n-1) + R_0) \quad (7)$$

Thus, $\{\hat{\Delta}_o^r(n),\ n \geq 2\}$ can be generated iteratively; computation of the statistics of $\hat{\Delta}_o^r(n)$ requires only the statistics of $\Delta_o^r(1)$ and that of $\hat{\Delta}_o^r(n-1)$ computed in the previous iteration. Let $F_{\Delta_o^r(n)}(x)$ (respectively, $F_{\hat{\Delta}_o^r(n)}(x)$) denote the cumulative probability distribution of $\Delta_o^r(n)$ (respectively, $\hat{\Delta}_o^r(n)$). Then the following holds.

Proposition 4.2 *For* $n \geq 1$,

$$\lim_{\lambda \to \infty} |F_{\Delta_o^r(n)}(x) - F_{\hat{\Delta}_o^r(n)}(x)| = 0$$

4.2. The Marginal Process, $\{\nu(n),\ n \geq 0\}$

Recall that $\nu(n) = P(\chi_o(n) < A)$, the fraction of nodes which get localised by iteration n. Since all the sensors need to get localised it is important to study $\{\nu(n),\ n \geq 0\}$.

Proposition 4.3 $\nu(1) = p + (1-p)(1 - e^{-\lambda p 2R_0})$ *and*

$$\lim_{\lambda \to \infty} (p + (1-p)(1 - e^{-\lambda p n 2R_0}) - \nu(n)) = 0 \quad (8)$$

5. RESULTS AND DISCUSSION

Figure 3 shows the variation of average localisation error (normalised to A) with the beacon density p for increasing iterations when $\lambda = 2$ per m. A is obtained as follows. In the simulation, we generate 1000 Poisson points for the random sensor placement model. The initial uncertainly for each sensor is then the expected length of this placement, i.e., $A = \frac{1000}{\lambda}$ and the initial location is its center. $E\chi_o(1)$ and $E\hat{\chi}_o(2)$ are also obtained from the previous analyses (Section 4.1). Note from Figure 3 that, in the case of $E\chi_o(1)$ the simulation results perfectly match the analysis. For $E\chi_o(2)$, the analytical result gives a fairly good match even for a low value such as $\lambda = 2$. For higher densities, analytical values match extremely well with those obtained by simulation ([4]). This suggests that (7) may be used to characterise the localisation errors iteratively for values of intensity of practical interest. Figure 3 also shows the benefits of the iterative scheme. When 10% sensors are beacons ($p = 0.1$), the localisation error is reduced from 40% in the first iteration to less than 5% in 20^{th} iteration.

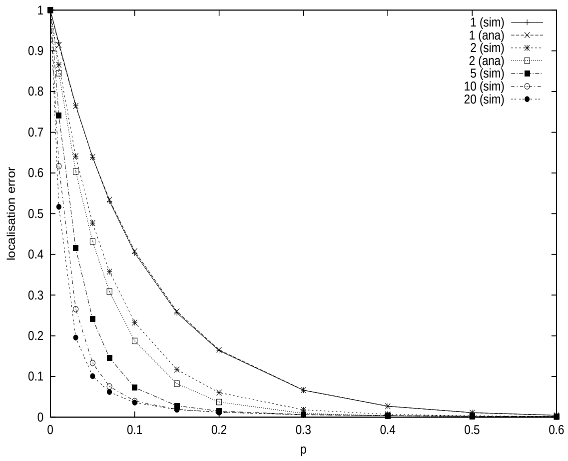

Fig. 3. Normalised localisation error vs. p for $\lambda = 2$ per m, $R = 2$ m, $A = 200$m. Analysis and Simulation.

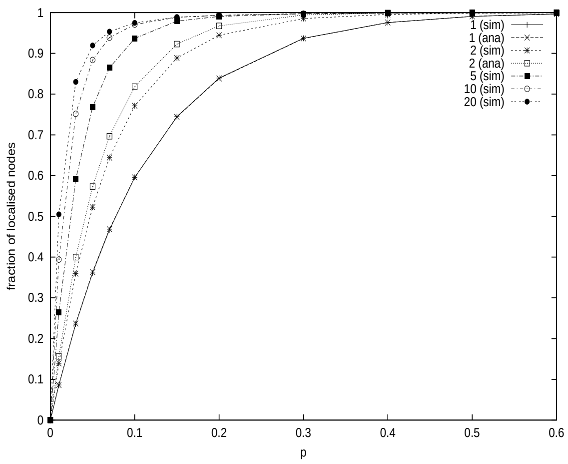

Fig. 4. Fraction of localised nodes vs. p for $\lambda = 2$ per m, $R = 2$ m. Analysis and Simulation.

Algorithm 1 Localisation Algorithm on the real line

if beacon **then**
  $X_i = (v_i, v_i)$
  broadcast (i, X_i)
else
  initialise $v_i = 0$, $x_i^l = -A/2$, $x_i^r = A/2$ and $X_i = (x_i^l, x_i^r)$
  upon receiving a broadcast from node k, (k, X_k), set $X_i = (\max(x_i^l, x_k^l - R_0), \min(x_i^r, x_k^r + R_0))$
  broadcast (i, X_i)
  $v_i = \frac{x_i^l + x_i^r}{2}$

Figure 4 shows the variation of $v(n)$ with p when $\lambda = 2$ per m. $v(1)$ and $v(2)$ are also obtained analytically (see Section 4.2). Observe that $(p + (1-p)(1 - e^{-\lambda p n 2R_0})$ is an upper bound on $v(n)$ for $n \geq 2$. This follows from the analysis ([4]). On the other hand, for higher densities, the analytical and simulation results are in excellent match. Note from Figure 4 that with 10% beacons, starting with about 60% nodes getting localised in the first iteration, by 20^{th} iteration about 98% the nodes are localised whereas with 20% beacons, starting with about 80% localised nodes in the first iteration, by 20^{th} iteration almost all the nodes are localised and the localisation error is less than 1% (Figure 3). Note that, with iterations the gains are reduced since since the number of nodes (hence beacons) is finite and every iteration fetches information about reachable beacons one more hop away to a sensor.

It is important to understand that effectiveness of IR does not depend on λ being high. Note from Proposition 4.1 that for a given p, localisation errors will be small if $\rho := \lambda 2R_0$ is large. Similarly, the fraction of sensors which get localised per iteration will be large for large ρ as Proposition 4.3 shows. Therefore, even in networks which are not very dense, IR will be an effective method as long as ρ is large.

6. LOCALISATION ALGORITHM

The iterative scheme in (3-4) is easy to implement in sensor networks as a distributed asynchronous algorithm 1. In Algorithm 1 each sensor locates itself to the center of its localisation set. Figure 5 shows the variation of location estimation error (normalised to range) with p using Algorithm 1 for increasing iterations. Observe that by choosing parameter p appropriately accuracy can be traded with the cost according to the needs of the application.

7. CONCLUSION

We proposed an extremely simple method of localisation that relies only on a basic communication capability of sensors. The method is rendered efficient because a sensor, after localising itself, aids others in localisation. We analysed the localisation process on the line and gave an iterative method of computing the average localisation error and the fraction of sensors localised. The results show that if the number of neighbours of a sensors are sufficiently high, the sensors can localise themselves collaboratively with small error using only a few beacons among them.

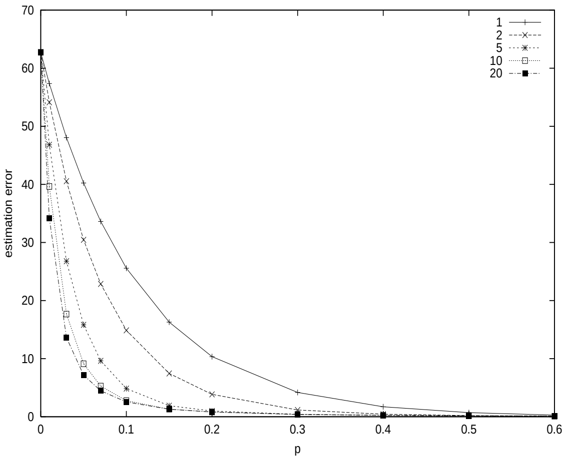

Fig. 5. Normalised location estimation error vs. p for $\lambda = 2$ per m, $R = 2$ m.

8. REFERENCES

[1] A. Savvides et al. Dynamic Fine-Grained Localization in Ad Hoc Networks of Sensors. In *ACM SIGMOBILE*, 2001.

[2] R. Nagpal et al. Organising a Global Coordinate System from Local Information on an Ad Hoc Sensor Networks. In *2nd Intl. Workshop on Information Processing in Sensor Networks (IPSN)*, 2003.

[3] J. Gibson. *The Mobile Communications Handbook*. IEEE Press, 1999.

[4] A. Karnik. *Optimal Self-organisation of Ad Hoc Wireless Sensor Networks*. PhD thesis, Indian Institute of Science, 2004.

[5] D. Niculescu and B. Nath. Ad Hoc Positioning System (APS). In *IEEE INFOCOM*, 2003.

[6] S. Simic and S. Sastry. Distributed Localization in Wireless Ad Hoc Networks. Memo. No. UCB/ERL M02/26, 2002.