Evaluating Parameters for Localization
In Wireless Sensor Networks

Mayank Gupta mayank19005@gmail.com
ECE Department, Lovely Professional University, India

Abstract: Advances in technology have made it possible to build ad hoc sensor networks using inexpensive nodes consisting of a low power processor, a modest amount of memory, a wireless network transceiver and a sensor board. Many novel applications are emerging: habitat monitoring, smart building failure detection and reporting, and target tracking. In these applications it is necessary to accurately orient the nodes with respect to a global coordinate system in order to report data that is geographically meaningful. Ad hoc sensor networks present novel tradeoffs in system design. On the one hand, the low cost of the nodes facilitates massive scale and highly parallel computation. On the other hand, each node is likely to have limited power, limited reliability, and only local communication with a modest number of neighbors. These application contexts and potential massive scale make it unrealistic to rely on careful placement or uniform arrangement of sensors.

Localisation is defined as process of determining the geographical location of a sensor in the network. Localization algorithms can be divided into two categories: range-based and range-free. In range-based algorithms, nodes estimate their distance to seeds using some specialized hardware. The measurements are used in methods like triangulation or tri-lateration, which are based on the idea that a node location is uniquely specified when at least the coordinates of three reference points are available for a node. Although the use of range measurements results in a fine grained localization scheme, range-based algorithms require the sensors contain hardware to make range measurements. Range-free algorithms do not use radio signal strengths, angle of arrival of signals or distance measurements and do not need any special hardware. Range-free algorithms require that each node knows:

(a) which nodes are within radio range
(b) their location estimates.
(c) the (ideal) radio range of sensors.

The localization problem gives rise to two important hardware problems. The first, the problem of defining a coordinate system, the second, which is the more technically challenging, is the problem of calculating the distance between sensors (the ranging problem). The goal of localization is to determine the physical coordinates of a group of sensor nodes. These coordinates can be global, meaning they are aligned with some externally meaningful system like GPS, or relative, meaning that they are an arbitrary “rigid transformation” (rotation, reflection, translation) away from the global
Beacon nodes (also frequently called anchor nodes) are a necessary prerequisite to localizing a network in a global coordinate system. Beacon nodes are simply ordinary sensor nodes that know their global coordinates a priori. This knowledge could be hard coded, or acquired through some additional hardware like a GPS receiver. At a minimum, three non-collinear beacon nodes are required to define a global coordinate system in two dimensions.

(I) Centralized Vs Distributed:
In centralized localization, the positions of all agents are determined by a central processor. This processor gathers measurements from anchors as well as agents and computes the positions of all the agents. Centralized algorithms are usually not scalable and thus impractical for large networks. In distributed localization, such as GPS, there is no central controller and every agent infers its own position based only on locally collected information. Distributed algorithms are scalable and thus attractive for large networks.

(II) Absolute Vs Relative:
Absolute localization refers to localization in a single predetermined coordinate system. Relative localization refers to localization in the context of one’s neighbors or local environment hence, the coordinate system can vary from node to node.

(III) Non-cooperative Vs Cooperative:
In a non-cooperative approach, there is no communication between agents, only between agents and anchors. Every agent needs to communicate with multiple anchors, requiring either a high density of anchors or long-range anchor transmissions. In cooperative localization, inter-agent communication removes the need for all agents to be within communication range of multiple anchors; thus high anchor density or long-range anchor transmissions are no longer required. Since agents can obtain information from both anchors and other agents within communication range, cooperative localization can offer increased accuracy and coverage.

Outdoor and Indoor Localization:

Outdoor Localization: Examples of outdoor systems include GPS, LORAN-C, and radio-location in cellular networks. GPS is a distributed, absolute, and non-cooperative localization approach, relying on TOA estimates from at least four anchors (GPS satellites) to solve a four dimensional nonlinear problem (three spatial dimensions and one time dimension, since the agent is not synchronized to the anchors). Assisted GPS is a centralized version of GPS, reducing the computational burden on the agents. LORAN-C is a terrestrial predecessor of GPS, which offers centralized, absolute, and non-cooperative localization services based on TDOA. Cell phone radio-location services such as E911 commonly employ TDOA and are centralized, absolute, and non-cooperative.

Indoor Localization: Existing & emerging indoor localization methods include WiFi, radio-frequency identification (RFID), and UWB localization. RADAR, based on WiFi fingerprinting at multiple anchors; PlaceLab, using connectivity from 802.11 access points; and GSM base stations employ centralized, absolute, and non-cooperative approaches. Passive RFID
tags can be used in conjunction with RFID readers to provide connectivity-based localization that is centralized, relative, and non-cooperative. Both WiFi and RFID systems suffer from poor accuracy due to coarse measurements. On the other hand, UWB signals have a number of characteristics that make them more attractive for indoor localization, as well as for indoor communication in general. The fine delay resolution of UWB signals is well suited for estimating propagation times (e.g., for RTOA or AOA), since the performance of delay estimation algorithms improves with increasing transmission bandwidth. Moreover, the wide bandwidth allows multipath components to be resolved and enables superior signal penetration through obstacles.

Parameters for Localization:

Received Signal Strength Indication (RSSI):

In theory, the energy of a radio signal diminishes with the square of the distance from the signal’s source. As a result, a node listening to a radio transmission should be able to use the strength of the received signal to calculate its distance from the transmitter. In practice, however, RSSI ranging measurements contain noise on the order of several meters. This noise occurs because radio propagation tends to be highly non-uniform in real environments. For instance, radio propagates differently over asphalt than over grass. Physical obstacles such as walls, furniture, etc. reflect and absorb radio waves.

Radio Hop Count:

The key observation is that if two nodes can communicate by radio, their distance from each other is less than $R$ with high probability, where $R$ is the maximum range of their radios, no matter what their signal strength reading is. Thus, simple connectivity data can be useful for localization purposes. In particular, many groups have found “hop count” to be a useful way to compute inter-node distances. The local connectivity information provided by the radio defines an unweighted graph, where the vertices are sensor nodes, and edges represent direct radio links between nodes. The hop count $h_{ij}$ between sensor nodes $s_i$ and $s_j$ is then defined as the length of the shortest path in the graph between $s_i$ and $s_j$. Naively, if the hop count between $s_i$ and $s_j$ is $h_{ij}$ then the distance between $s_i$ and $s_j$, $d_{ij}$, is less than $R \cdot h_{ij}$, where $R$ is again the maximum radio range.

Time Difference of Arrival (TDoA):

Time Difference of Arrival (TDoA) is a commonly used hardware ranging mechanism. In TDoA schemes, each node
is equipped with a speaker and a microphone. Some systems use ultrasound while others use audible frequencies. In TDoA, the transmitter first sends a radio message. It waits some fixed interval of time, \( t_{delay} \) (which might be zero), and then produces a fixed pattern of “chirps” on its speaker. When listening nodes hear the radio signal, they note the current time, \( t_{radio} \), then turn on their microphones. When their microphones detect the chirp pattern, they again note the current time, \( t_{sound} \). Once they have \( t_{radio}, t_{sound}, \) and \( t_{delay} \), the listeners can compute the distance \( d \) between themselves and the transmitter using the fact that radio waves travel substantially faster than sound in air.

\[ d = (s_{radio} - s_{sound}) \cdot (t_{sound} - t_{radio} - t_{delay}) \]

TDoA methods are impressively accurate under line-of-sight conditions. However, they perform best in areas that are free of echoes, and when the speakers and microphones are calibrated to each other. Several groups are working to compensate for these issues, which will likely lead to even better field accuracy.

The downside of TDoA systems is that they inevitably require special hardware to be built into sensor nodes, specifically a speaker and a microphone. TDoA systems perform best when they are calibrated properly, since speakers and microphones never have identical transmission and reception characteristics. Furthermore, the speed of sound in air varies with air temperature and humidity which introduces inaccuracy into equation. Finally, the line-of-sight constraint can be difficult to meet in some environments.

**Angle of Arrival (AoA):**

Some algorithms depend on angle of arrival (AoA) data. This data is typically gathered using radio or microphone arrays, which allow a listening node to determine the direction of a transmitting node. It is also possible to gather AoA data from optical communication methods. In these methods, several (3-4) spatially separated microphones hear a single transmitted signal. By analyzing the phase or time difference between the signal’s arrival at different microphones, it is possible to discover the angle of arrival of the signal. These methods can obtain accuracy to within a few degrees. Unfortunately, angle-of-arrival hardware tends to be bulkier and more expensive than TDoA ranging hardware, since each node must have one speaker and several microphones. Furthermore, the need for spatial separation between speakers is difficult to accommodate as the form factor of sensors shrinks.

**Issues in Localization Algorithm Design**

**Resource Constraints:**

Sensor networks are typically quite resource-starved. Nodes have rather weak processors, making large computations infeasible. Moreover, sensor nodes are typically battery powered. This means communication, processing, and sensing actions are all expensive, since they actively reduce the lifespan of the node performing them. Not only that, sensor networks are typically envisioned on a large scale, with hundreds or thousands of nodes in a typical deployment. This fact has two important consequences: nodes must be cheap to fabricate, and trivially easy to
deploy. Nodes must be cheap, since fifty cents of additional cost per node translates to for a one thousand node network. Deployment must be easy as well: thirty seconds of handling time per node to prepare for localization translates to over eight man-hours of work to deploy a 1000 node network. Localization is necessary to many functions of a sensor network; however, it is not the purpose of a sensor network. Localization must cost as little as possible while still producing satisfactory results. That means designers must actively work to minimize the power cost, hardware cost, and deployment cost of their localization algorithms.

**Node Density:**
Many localization algorithms are sensitive to node density. For instance, hop count based schemes generally require high node density so that the hop count approximation for distance is accurate. Similarly, algorithms that depend on beacon nodes fail when the beacon density is not high enough in a particular region. Thus, when designing or analyzing an algorithm, it is important to notice the algorithm’s implicit density assumptions, since high node density can sometimes be expensive if not totally infeasible.

**Non-convex Topologies:**
Localization algorithms often have trouble positioning nodes near the edges of a sensor field. This artifact generally occurs because fewer range measurements are available for border nodes, and those few measurements are all taken from the same side of the node. In short, border nodes are a problem because less information is available about them and that information is of lower quality. This problem is exacerbated when a sensor network has a non-convex shape: Sensors outside the main convex body of the network can often prove un-localizable. Even when locations can be found, the results tend to feature disproportionate error.

**Environmental obstacles and terrain irregularities:**
Environmental obstacles and terrain irregularities can also wreak havoc on localization. Large rocks can occlude line of sight, preventing TDoA ranging, or interfere with radios, introducing error into RSSI ranges and producing incorrect hop count ranges. Deployment on grass vs. sand vs. pavement can affect radios and acoustic ranging systems. Indoors, natural features like walls can impede measurements as well. All of these issues are likely to come up in real deployments, so localization systems should be able to cope.

**Self-aware Wireless Sensor Network**
It has gained widespread interest in the recent years, especially with the advent of the ZigBee Alliance standard based on direct sequence spread spectrum (DSSS) communications. Its usage has spanned across a vast array of applications, for detecting, identifying, localizing, monitoring, or tracking purposes, such as in smart-home network, mobile healthcare, asset/target tracking, military or security surveillance, environmental monitoring, natural-catastrophe detection, as well as industrial control and automation. In these applications, the sensed data collected by the sensor nodes are often only meaningful if the location where the data come from are known. A straightforward way to do this is by
manual positioning with the aid of a central system administrator. However this can sometimes be impractical when there are a large number of sensor nodes involved in the network. An alternative way is by simply using the Global Positioning System (GPS). But this is often not an effective solution especially when the cost, size or power of the nodes are of a concern. Not only that, the use of the GPS is not always feasible, such as when in an indoor, urban or forest environment, where there is no clear line of sight between the receivers and the satellites. These hence give rise to the introduction of self-aware wireless sensor network where the nodes, unlike those intervened by the aid of an administrator or GPS, could self-localise themselves by working together cooperatively in a peer-to-peer manner so as to form a geographical map of the network.

Cooperative localisation is an emerging paradigm. Each of the sensor nodes in the network are designed to cooperate with its neighbouring nodes iteratively until all the nodes arrive at a consensus about their location in a particular coordinate system. Depending on the applications, there are two types of self-aware sensor network systems, one that is based on anchor-free localisation, and the other that is based on anchor-aided localisation. The difference is due to the presence of anchor nodes in the sensor network system. Now anchor nodes are reference nodes placed in the network with self-known geographical coordinates obtained either via a system administrator or GPS. Example: of a randomly distributed anchor-free wireless sensor network. Since there is no reference coordinate in the system, the sensor node’s localisation within its locality is established based on an arbitrary relative coordinate system in the context of its surrounding neighbours. In contrast, depicts an example of a anchor-aided sensor network system. With the presence of a reference coordinate, absolute localisation can hence be established at each of the sensor nodes based on a pre-specified coordinate system. Note that such absolute localisation can also work even when the reference nodes are not in the immediate vicinity of the node. Due to the cooperative nature of the sensor network, the node is able to adjust and propagate its localisation estimate in accordance with what its neighbouring nodes have observed in their localities. Notice that these anchor nodes need not necessarily be the sensor nodes themselves. In some cases, it might be useful to have the central receiver, which is basically the information sink for the data gathered from the network, to act as the source of reference for the basis of absolute localisation. It is also worthwhile to note that, besides internal localization among the sensor nodes, the localisation can also be applied externally to any particular target-of-interest that has entered within the compound of the network based on

Cooperative localisation in anchor-free sensor network.
either the relative or absolute coordinate system.

However, active research remains in the design of the localisation algorithms that are robust against the node's relative motion, the sensor orientation, as well as the other environmental uncertainties such as multipaths, interferences, and noise. The relative motion between the nodes, as a result of (a) stationary anchor nodes and moving sensor nodes, (b) moving anchor nodes and stationary sensor nodes, and (c) moving anchor nodes and moving sensor nodes, has previously been studied and investigated. Little work has however been conducted to address the orientation of the nodes, especially in a randomly deployed wireless sensor network application. This is critical as some applications, such as the wireless surveillance sensor system, need to know the sensor nodes' angles of orientation in their operations. Besides that, the received signal strength can also be readily affected by mismatched signal polarisation between the arbitrarily orientated transmitting and receiving ends.

**APIT (Approximate Point in Triangle):**

APIT is quite a bit different from the beacon-based distributed algorithms described so far. APIT uses a novel area-based approach, in which nodes are assumed to be able to hear a fairly large number of beacons. However, APIT does not assume that nodes can range to these beacons. Instead, a node forms some number of “beacon triangles”, where a beacon triangle is the triangle formed by three arbitrary beacons. The node then decides whether it is inside or outside a given triangle by comparing signal strength measurements with its nearby non-beacon neighbors. Once this process is complete, the node simply finds the intersection of the beacon triangles that contains it. The node chooses the centroid of this intersection region as its position estimate. An example of this process: each of the triangles represents a triple of beacons and the intersection of all the triangles defines the position of the unknown node.

*The actual algorithm is as follows:*

**Step 1:** Receive beacon positions from hearable beacons.
**Step 2:** Initialize inside-set to be empty.
**Step 3:** For each triangle $T_i$ in possible triangles formed over beacons, add $T_i$ to inside-set if node is inside $T_i$.
**Step 4:** Compute position estimate as the center of mass of the intersection of all triangles in inside-set.

The point in triangle (PIT) test is based on geometry. For a given triangle with points A, B, and C, a given point M is outside triangle ABC, if there exists a direction such that a point adjacent to M is further/closer to points A, B, and C simultaneously.
Node position estimated as the center of mass of the intersection of a number of beacon triangles for which a given node is inside.

Otherwise, M is inside triangle ABC. Unfortunately, given that typically nodes cannot move, an approximate PIT (APIT) test is proposed which assumes sufficient node density for approximating node movement. If no neighbor of M is further from/closer to all three anchors A, B, and C simultaneously, M assumes that it is inside triangle ABC. Otherwise, M assumes it resides outside this triangle. This algorithm is described as being range-free, which means that RSSI range measurements are required to be monotonic and calibrated to be comparable but are not required to produce distance estimates. It could be that the effort put into RSSI calibration would produce an effective enough ranging estimate to be useful for gradient techniques, making the range-free distinction potentially moot. The APIT algorithm also requires a relatively high ratio of beacons to nodes, requires longer range beacons, and is susceptible to erroneously low RSSI readings.

**Trilateration:**

In the following we will show the development of the formula for determining the position using three non-linear points and the distances to these points.

This formula can be used in different situations, e.g., for determining the motes coordinates in the initialization phase using the buzzer motes, or for computing the position of the moving target using the distance information derived from the light values. We assume, that we have given the following values:

- the coordinates of three reference points $B_0: (x_0, y_0)$, $B_i: (x_i, y_i)$
- the distance from a point M to each of the reference points: $d_i$: distance from $M$ to $B_i$ for $i \in \{0,1,2\}$

**Determining Location by Trilateration**

We are looking for the coordinates of M. Let $(x, y)$ be the coordinates to be determined for M. For solving this problem, we exploit the act, that lies on the circles defined by the center $B_i$ with radius $d_i$. We can therefore use the equations defined for these each of these circles, given as:

$$(x - x_i)^2 + (y - y_i)^2 = d_i^2$$

First, we transform the equation by multiplying the squares on the left-hand-side:

$$x^2 - 2xx_i + x_i^2 + y^2 - 2yy_i + y_i^2 = d_i^2$$
We can further transform these formulas by introducing constants resulting in:

\[ x^2 - 2xx_i + y^2 - 2yy_i + c_1 = d_i^2 \]

We can now simply obtain the values for \( x \) and \( y \) by applying basic arithmetic transformation steps; in the course of these transformations, the assumption about the relative positions of the reference points have to be applied; these assumption can directly be derived from those cases that would result in a division-by-zero situation.

**CONCLUSION**

Location-awareness is a key feature of future-generation wireless networks, enabling a multitude of applications in the military (e.g., blue force tracking), public (e.g., search and-rescue), and commercial (e.g., navigation) sectors. Cooperation among nodes has the potential to dramatically improve localization performance. In this paper I have evaluated the various parameters that concern the localisation in a wireless sensor network. With the advent of new technologies, it will be possible to use same algorithm to compute localisation for static as well as mobile WSNs. One such has been developed by Suprakash Dutta et al.

**References:**


