

LOCALIZATION OF WIRELESS SENSOR NETWORK

Range free anchor-based algorithm using Monte Carlo Localization

Prof. Usha Neekeleetan¹, Princess Mariam Zawu².

Head of Department, of EC L D Engineering College, Ahmedabad, Gujarat, India¹

PG Student [ECS], Dept. of EC, L.D College of Engineering, Ahmedabad, Gujarat, India²

Abstract - *Abstract*— recent proceedings in radio and embedded systems have enabled the increase of wireless sensor networks. Wireless sensor networks are tremendously being used in different environment to perform various monitoring tasks such as search, rescue, disaster relief, target tracking and a number of tasks in smart environments. In many of those tasks, node localization is inherently one of the system parameters. Node localization is required to report the origin of events, assist in group querying of sensors, routing and to and also to know the answer of network coverage. So, one of the fundamental challenges in wireless sensor is node localization. This paper presents an accurate range-free localization scheme for nodes in mobile wireless sensor networks. As it is already known that the sequential Monte Carlo localization method works well for localization in mobile WSNs. Based on the sequential Monte Carlo method, the TSBMCL algorithm utilizes the nodes for localization.

Keywords—*wireless sensor networks; mobile WSNs; localization; TSBMCL; Mobile Location; Monte Carlo*

1. INTRODUCTION

The massive advances of microelectromechanical systems (MEMS), computing and communication technology have fomented the emergence of massively distributed, wireless sensor networks consisting of hundreds and thousands of nodes. Each node is able to sense the environment, perform simple computations and communicate with its other sensors or to the central unit. One way of deploying the sensor networks is to scatter the nodes throughout some region of interest. This makes the network topology random. Since there is no a priori communication protocol, the network is ad hoc. These networks are tremendously being implemented to perform a number of tasks, ranging from environmental and natural habitat monitoring to home networking, medical applications and smart battlefields. Sensor network can signal a machine malfunction to the control center in a factory

or it can warn about smoke on a remote forest hill indicating that a forest fire is about to start. On the other hand wireless sensor nodes can be designed to detect the ground vibrations generated by silent footsteps of a burglar and trigger an alarm. Since most applications depend on a successful localization, i.e. to compute their positions in some fixed coordinate system, it is of great importance to design efficient localization algorithms. In large scale ad hoc networks, node localization can assist in routing. In the smart kindergarten node localization can be used to monitor the progress of the children by tracking their interaction with toys and also with each other. It can also be used in hospital environments to keep track of equipment, patients, doctors and nurses. For these advantages precise knowledge of node localization in ad hoc sensor networks is an active field of research in wireless networking. Unfortunately, for a large number of sensor nodes, straightforward solution of adding GPS to all nodes in the network is not feasible because: In the presence of dense forests, mountains or other obstacles that block the line-of-sight from GPS satellites, GPS cannot be implemented. [1] The power consumption of GPS will reduce the battery life of the sensor nodes and also reduce the effective lifetime of the entire network. [2] In a network with large number of nodes, the production cost factor of GPS is an important issue. [3] Sensor nodes are required to be small. But the size of GPS and its antenna increases the sensor node form factor. For these reasons an alternate solution of GPS is required which is cost effective, rapidly deployable and can operate in diverse environments. The two environments were one that allows preplace anchor nodes when distribution is ready and has high accuracy and the other is one that those not allow preplace and requires accuracy of about 30%[5]. At present, most of the localization

algorithms are designed for the static wireless sensor network. If the static sensor network localization algorithm is applied to the mobile sensor network, a series of problems will be caused, for instance, the accuracy of localization will be reduced because of the mobility of nodes, the node energy consumption is accelerated, etc. Some proposals are emerged in terms of the node localization of the mobile wireless sensor network. In the literature [6], this paper verifies the correction and reliability of ROA (Rectangle Overlapping Approach). They also made the simulation of the ROA algorithm by using the Borland c++ builder 6.0 program languages. The researchers also discussed the traverse strategy of a moving beacon, which is moving in horizontal and vertical directions and moving in random direction. The paper also discussed the three important values, the sensing range of a circle and a beacon moving each direction and the angle by the directional antenna. The researchers also explored the effects among each of the three factors through simulations. In the literature [2], the author puts forward the TSBMCL algorithm according to the MCB algorithm. On account of the insufficient anchor nodes near the unknown nodes, the ordinary nodes with the good locations are screened as the assistant position of the temporary anchor nodes. In the algorithm, it is able to satisfy the location requirement of mobile WSN with the low cost and high accuracy. The sampling procedure is complicated for the classic Monte Carlo localization algorithm while the sampling efficiency is low. This paper proposes a Monte Carlo localization algorithm in the combination of hop count/distance transformation model. In this algorithm, it can avoid the direct use of node communication range to determine the sampling area and there is no requirement on the additional ranging hardware. A better location performance is achieved.

I. Localization using range-free anchor based algorithm using Monte Carlo localization

A range-free localization algorithm for mobile sensor networks based on the Sequential Monte

Carlo method. The Monte Carlo method has been extensively used in robotics where a robot estimates its localization based on its motion, perception and possibly a relearned map of its environment. Monte Carlo method as used in robotics to support the localization of sensors in a free, unmapped terrain. The authors assume a sensor has little control and knowledge over its movement, in contrast to a robot. Apart from the experiments with MCL, there are at the moment few localization protocols specifically designed with mobile wireless sensors in mind. Most of the papers presenting localization algorithms suggest that supporting mobility can be achieved by rerunning the localization algorithm after some time interval, either static or adaptable. While this is not optimal but feasible in some cases, the whole class of algorithms using information from distant nodes or iterative approaches will suffer from severe information decay. At the time the information reaches a distant node that wants to use it, it is very likely that the whole network configuration has changed. A node will therefore always calculate an inaccurate location, not due to the lack of information or to the intrinsic inaccuracy of the calculations it uses, but due to the way its localization algorithm gathers this information.

MONTE CARLO LOCALIZATION

1) Bayes filtering

The MCL represents the robot's positional certainty at an arbitrary location in a given grid map. A robot calculates the posterior probability by using the Bayes filter [9] based on the odometry and range data as follows:

$$Bel(x_t) = x_t Z_{0:t} U_{0:t} \quad (1)$$

Where x_t denotes the robot pose $(x, y, 0)$ at time t , $z_{0:t} = \{z_0, z_1, \dots, z_t\}$ denotes the measurements of the range sensor (e.g. Laser scanner, sonar, IR sensor, etc.) up to t , and $u_{0:t} = \{u_0, u_1, \dots, u_t\}$ is the

odometric data from the wheel encoder.

Reliability of measurements varies with the accuracy of the range sensor. In order to cope with various uncertainties, probability models are used to reflect the errors: sensor model (or perception model) and motion model (or action model). The Bayes filter is conducted in two steps (i.e., prediction and update), which can be represented by the following equation: $Bel(x_t) = \frac{1}{q} \int p(z_t | x_t) JP(x_t | x_{t-1}, u_t) Bel(x_{t-1}) dx_{t-1}$ where q is the normalizing constant, $p(x_t | x_{t-1}, u_t)$ is the motion model, and $p(z_t | x_t)$ is the perception model. We assume that both the motion and sensor models are described by a Gaussian distribution and the noise is modelled by the Gaussian noise with zero mean.

1) Particle filters

The particle filter used in MCL represents a posterior distribution $p(x_t | z_{0:t}, u_{0:t})$ by a set of random samples drawn from this distribution. Each particle corresponds to the robot pose (x, y, θ) . Among several variants of the particle filter, the SIR (Sampling Importance Resampling) algorithm is adopted in this research [11]. The approach is composed of the next three steps; sampling, importance weighting and resampling. In sampling, the new sample set X_t , is generated according to the motion model $p(x_t | x_{t-1}, u_t)$ from the past sample set X_{t-1} distributed by $Bel(x, t)$. In importance weighting, the importance factor $w_t(i)$ is evaluated using the sensor model, where q is a normalization constant. $p(z_t | x_t)$ is calculated using the similarity measure function is. The positional probability is computed by the evaluation function, which calculates range differences between the laser scan data Z_t and the expected reference range data $x_t(i)$ computed from a grid map. In resampling, the new sample set X_t is randomly chosen from X_t according to the distribution defined by importance factor (w_t). The importance factor $w_t(i)$ of the sample set X_t , at time t is initialized to $1/N$. Through the recursive computation of three steps, the samples converge to the pose with highest probability.

B). TOPOLOGICAL INFORMATION

Topological Map building based on Thinning Algorithm. Topological information is abstraction of the environment in terms of the nodes representing discrete places and the edges connecting them together. The topological information can be generated either by the GVG method or the thinning method. The GVG method is robust to various environments and can be extended to the higher-dimensional space. However, the map creates the boundary edges and weak meet points which are unnecessary in navigation. On the other hand, the thinning method does not create such information and is robust to sensor noise and various environments because it is based on the probabilistic framework. A thinning method is one of the image processing algorithms which are used to detect the skeleton of images. Fig. 1 illustrates the concept of thinning. The objects on the left can be described satisfactorily by the structure composed of connected lines (i.e., 'T' shape drawn with thin lines on the right). Note that connectivity of the structure is still preserved even with thin lines. In the case of mobile robots, the connected lines are used as paths on which a robot travels without colliding with other objects.

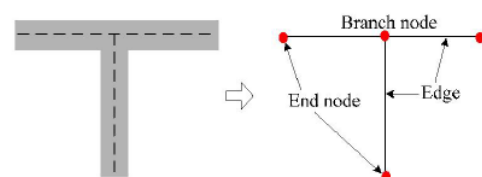


Fig. 1. Extraction of topological edges and nodes by thinning algorithm.

II. Model

Temporary-seed based Monte Carlo localization (TSBMCL), including two main parts. The first is the process of voting temporary seeds, and the second one is the process that the temporary seeds aid other nodes for localization. The whole TSBMCL

algorithm is given as follows: The procedure of voting temporary seeds is depicted below.

- (a). All the seeds who are always aware of their locations flood information in the whole network, the data frames include (xseed,yseed),ID Seed, TTL, denotes the coordinate value of the seed, ID seed denotes the seed's ID number, and TTL denotes the maximum flooding hops initialized by 2.
- (b). All the nodes who need localization record the coordinate value and the ID number of the seeds whose flooding information they can receive. If the TTL value is 2, then the node set it 1 and forward this data frame to other nodes. If the TTL is 1, the node sets it 0 and never forwards it. Thus all the seeds' data frames could be forwarded once at most.

After the phase of seeds' information flooding, every node gets an anchor box under the algorithm of MCB, then it could further get its weight by which the node could judge whether it could become a temporary seed or not. Localization would be realized.

The formula for weight computation is given as follows:

$$TA = \alpha \times \frac{N_{nos}}{S_{anchorbox}}$$

In formula (1),TA denotes the weight value,S_{anchorbox} denotes the area of the node's anchor box, N_{nos} denotes the number of seeds within two hops, α is a factor for adjusting the weight's scaling. Meanwhile, two thresholds should be set. The upper limit is TA_{up}, and the lower limit is TA_{down}. If the TA Value of one node exceeds TA_{down}, and then the node is localized under the algorithm of MCB. If the TA value Exceeds TA_{up}, then this node becomes a temporary seed. If the TA value is less than TA_{down}, then this

node is not localized and it waits for the flooding information of the temporary seeds.

III. Performance

In real world applications, we often have to improve the localization accuracy in some parts of the whole monitoring region according to the position where the emergency happens. For example, when fire takes place in the forest demonstrates that the TSBMCL scheme could effectively optimize the localization situation in the mobile WSNs.

The localization failure rate of MCL/TI was significantly lower than that of standard MCL in Fig. 5 when the number of samples was identical. For example, when 1000 samples were used, the failure rate of 6% for MCL/TI is much lower than that of 25% for the standard MCL. The reason for the superior performance of the proposed scheme is that the sample density of MCL/TI is much higher than that of the standard MCL. Also, the localization failure rate of MCL using 3000 samples was similar to that of MCL/TI using 900 samples. This result shows that the required number of samples can be reduced to 30% of the MCL samples for the typical office environment.

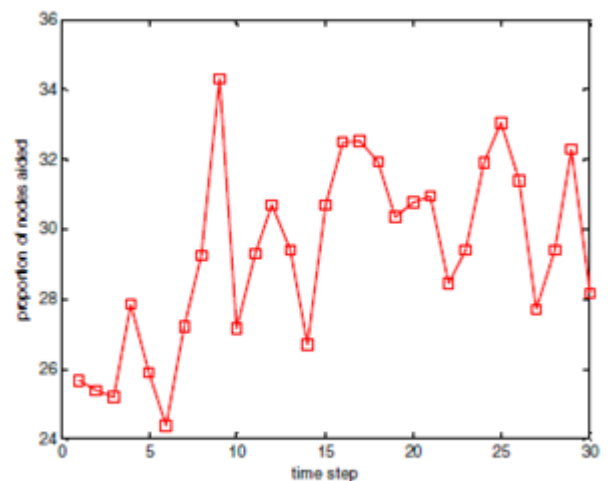


Figure 4. proportion of nodes aided

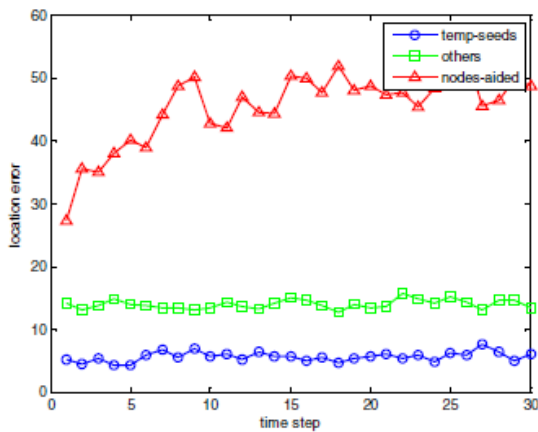


Figure 2. localization error

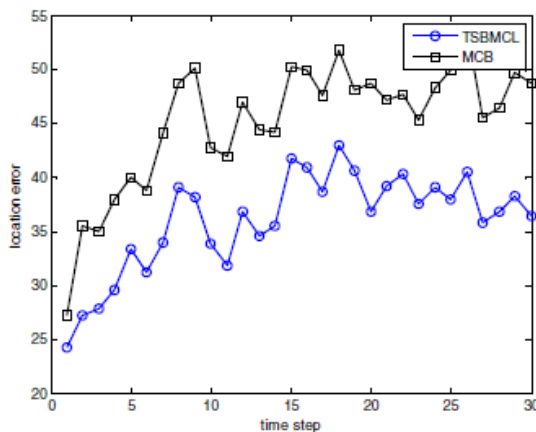


Figure 3. localization error

IV. Conclusion

This paper presented an accurate, range-free localization schemes for nodes in mobile wireless sensor networks.

As is already known, the sequential Monte Carlo localization method works well for localization in mobile WSNs. Based on the sequential Monte Carlo method, the TSBMCL algorithm utilizes the nodes which are localized well to aid other nodes for localization. Our simulation results show that the TSBMCL algorithm outperforms the MCB algorithm by improving the localization accuracy of the network.

V. References

[1] Ju Mei, Jinyan Gao, and Di Chen "Range-Free Monte Carlo Localization for Mobile Wireless Sensor Networks" *Journal of IEEE international conference on wireless sensor.*

[2] Tae-Bum Kwon, Ju-Ho Yang, Jae-Bok Song, Woojin Chung" Efficiency Improvement in Monte Carlo Localization Through Topological Information" *Journal of IEEE International conference of mobile wireless sensor.*

[3] Xi-Rong Bao, Shi Zhang, Ding-Yu Xue, 2008"Research on the Self-localization of Wireless Sensor Networks", *Journal of IEEE International conference on embedded software and system*

[4] YAO-HUNG WU AND WEI-MEI CHEN, 2009 "Localization Of Wireless Sensor Networks Using A Moving Beacon With Directional Antenna", *Journal of IEEE on high performance computing and computation*

[5] Younes Ahmadi, Naser Neda and Reza Ghazizade" *Range Free Localization in Wireless Sensor Networks for Homogenous and Non-Homogenous Environment* "