

3D Node Localization from Node-to-Node Distance Information using Cross-Entropy Method

Shohei Ukawa* Tatsuya Shinada Masanori Hashimoto† Yuichi Itoh Takao Onoye

Dept. Information Systems Engineering Osaka University

ABSTRACT

This paper proposes a 3D node localization method that uses cross-entropy method for the 3D modeling system. The proposed localization method statistically estimates the most probable positions overcoming measurement errors through iterative sample generation and evaluation. The generated samples are evaluated in parallel, and then a significant speedup can be obtained. We also demonstrate that the iterative sample generation and evaluation performed in parallel are highly compatible with interactive node movement.

Index Terms: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling; H.5.2 [Information Interfaces and Presentation]: User Interfaces—Input Devices and Strategies

1 INTRODUCTION

In recent years, services using 3D models are spreading widely thanks to advances in hardware and IT technology, and their demand has been increasing in various fields. Along with this, various high-performance 3D modeling software is commercially available. However, such software is designed for experienced people for professional usage, and hence it is too difficult for most of people to use. An intuitive and easy way of 3D modeling for non-expert people is demanded.

We devised a concept of three-dimensional shape recognition system that built a sensor network by embedding sensor nodes into deformable material such as clay, and we are working for actualizing this system[1][2]. In this iClay system, a number of 1mm³-class tiny sensor nodes are distributed in plastic clay and they organize a wireless sensor network. This system measures the distance between each sensor node and other sensor nodes located in the vicinity, collects the distance information via a sensor network, and reproduces the clay shape by estimating the relative position of each sensor node.

In this paper, we propose a method to localize each sensor node in a three dimensional space from the node-to-node distance information obtained through the sensor network. This work assumes that the sensor nodes and sensor network are available and the distance information is collected to a computer.

2 PROPOSED LOCALIZATION METHOD

2.1 Problem formulation

A sensor network that consists of N nodes is assumed. Now, a set of the node-to-node distance information is given, where d_{ij} is the distance between the i -th node to j -th node. This distance information includes measurement errors. Also, when the i -th node is distant from the j -th node, d_{ij} cannot be measured. In this case

*e-mail: ukawa.shohei@ist.osaka-u.ac.jp

†e-mail: hasimoto@ist.osaka-u.ac.jp

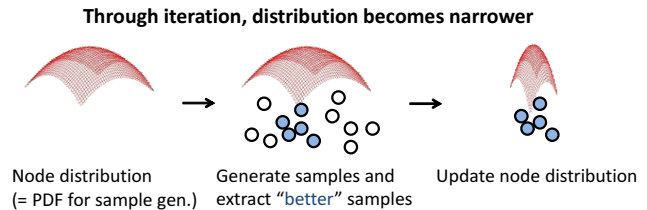


Figure 1: Solutions converge through iteration.

$d_{ij}=0$. Based on this distance information, we estimate the locations q_i of N nodes.

Due to the measurement errors included in the distance information, the exact locations cannot be computed since the correct locations and the measured distances are inconsistent. We would rather compute the most probable locations. In this context, we want to minimize the sum of localization errors between the true node position and estimated node position. We use dRMS to evaluate the errors in relative coordinates. The dRMS error is computed by comparing all the pairwise distances, and is defined as

$$dRMS^2 = \frac{1}{N^2} \sum_i^N \sum_j^N (||p_i - p_j|| - ||q_i - q_j||)^2, \quad (1)$$

where p_i is the true position of the i -th node in the absolute three-dimensional coordinates. q_i is also in the absolute three-dimensional coordinates. $||p_i - p_j||$ means the Euclidean distance between p_i and p_j . Now, the node localization problem corresponds to finding q_i which minimizes dRMS.

However, dRMS cannot be calculated during the node localization process because the true node position is unknown. Then, we use the given node-to-node distance information d_{ij} instead of $||p_i - p_j||$ while d_{ij} includes measurement errors.

$$eval^2 = \frac{1}{N^2} \sum_i^N \sum_{j, d_{ij} \neq 0}^N (d_{ij} - ||q_i - q_j||)^2. \quad (2)$$

2.2 Proposed method

The proposed method consists of two parts. The first part generates a set of initial solutions referring to [3], and they are given to the successive second part. The second part starts the iteration of cross-entropy method[4].

In the iteration loop of Figure 1, we first update the node distribution, which will be used as PDF for the sample generation. We now have a set of solutions. A solution consists of localization results of individual nodes q_i and its value of objective function. Through the iteration, we expect that the solutions will converge to the global optimal point. To guide the solutions, we select top NS_{top} good solutions from NS_{gen} solutions for the following computation, and construct the node distribution from the selected NS_{top} solutions. Here, Gaussian distribution is assumed for the node distribution.

We then generate NS_{gen} samples according to the obtained Gaussian distributions with the means and standard deviations. After that, we calculate the objective function of Eq. (2) for all the

samples. It is obvious that the sample generation and the computation of the objective function for each sample are independent of them for the other samples, and hence the cross-entropy based optimization method is highly compatible with parallel computing.

2.3 Extension to continuous real-time estimation

The proposed method in the previous section works well for a static object. On the other hand, this object transformation often happens since iClay system aims to reproduce the clay shape on the fly. During the object transformation, the node-to-node distance information is changing. Even if the iteration of the proposed method is kept going and new distance information is given, the proposed method cannot move the solutions since the Gaussian distributions are too narrow and the generated samples are not diverse enough.

To enable continuous real-time estimation, we need to expand the distributions of the random mechanism for exploring a larger solution space. To implement this distribution expansion, there are two problems; "when" and "how much" the distributions should be expanded.

To resolve these problems, we devised the following scheme empirically. First, the distributions are expanded when the following conditions are true: 1) The amount of increase in the objective function in the current iteration is A times or beyond larger than the amount of decrease in the previous iteration, where A is set to 10 in this work. 2) The objective function value needs to be smaller than the objective function value immediately after the previous distribution expansion.

Secondly, the distributions are expanded in the following expression. We add σ_{add}^2 to the variance of the distribution.

$$\sigma_{add}^2 = B(eval_k - eval_{k-1})^2, \quad (3)$$

where $eval_k$ is the objective function value at step k (current). Using the change in the objective function, i.e. the amount of shape transformation, the amount of distribution expansion is determined. B is empirically determined, and it is set to 400 in the experiments in the next section.

3 EXPERIMENTAL RESULTS

This section evaluates the node localization accuracy and efficiency of the proposed method. The proposed method is implemented with MATLAB. We used parallel computing toolbox for parallelize the sample generation and evaluation in the proposed method. The number of threads for parallel processing given to the toolbox was 80.

The experimental setup for clay and nodes is as follows. The clay volume is $50 \times 50 \times 50 \text{ mm}^3$, and $343(=7^3)$ nodes are distributed in the clay. It is assumed that each node can sense the distance to other nodes in a range of 20 mm. The distance information given to this experiment includes the measurement error generated according to [1].

3.1 Localization for static object

We localize the nodes in a static object with the proposed method, where a static object means its shape is unchanged. A rectangular solid in which nodes are uniformly distributed are considered.

Figure 2 shows the localized positions. The green circles represent true node positions, and the red arrows from these corresponds to the offsets between the true positions and localized positions. We can see that the nodes are well localized.

In order to minimize dRMS error, we indirectly minimize $eval$ since dRMS cannot be computed during the optimization as explained in Section 2.1. The traces of dRMS and $eval$ are highly correlated, which means the minimization of $eval$ is tightly related to dRMS minimization, as we expected. The optimization worked well thanks to cross-entropy method.

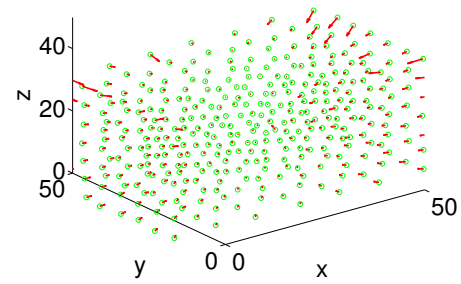


Figure 2: Estimated positions for a rectangular solid (static).

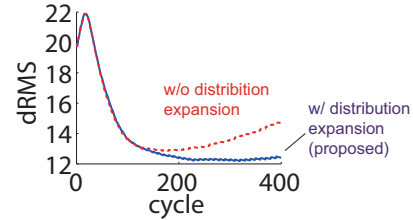


Figure 3: Node localization error dRMS for shape-transforming object from cube to sphere.

We next evaluate the impact of parallel computing thanks to the cross-entropy method. We executed the node localization for a cubic with and without parallel computing option. The CPU time reduction was significant and the speedup ratio was 22X.

3.2 Estimation for shape-transforming objects

Next, we show the node localization results for objects whose shape is changing, i.e. the distance information given to the proposed node localization method is being updated during the localization. We applied the proposed method to an object which was transforming from a cube to a sphere. The nodes were moved such that the clay volume was unchanged. In this experiment, the transformations were divided into 39 intermediate transformations and these transformations occurred during the 400 iterations in the proposed method.

Figure 3 shows dRMS variation though the iteration. We evaluated dRMS variation for two conditions; one is the proposed method that expands the distribution as explained in Section 2.3 and the other is the proposed method without the distribution expansion. We can see that dRMS keeps decreasing when the distribution expansion is applied. However, without the distribution expansion, the node localization method could not catch up with the shape transformation after 150 cycles. The distribution expansion defined by the two schemes worked well for tracking the shape transformation.

ACKNOWLEDGEMENTS

This work was partly supported by the SCOPE program of the Ministry of Internal Affairs and Communications.

REFERENCES

- [1] T. Shinada, M. Hashimoto, and T. Onoye, "Proximity Distance Estimation Based on Capacitive Coupling Between Imm³ Sensor Nodes," Proc. NEWCAS, 2013.
- [2] J. Kono, M. Hashimoto, and T. Onoye, "Feasibility Evaluation of Near-Field Communication in Clay with 1-mm³ Antenna," Proc. APMC, pp. 1121–1123, 2013.
- [3] R. Mautz, W. Ochieng, G. Brodin, and A. Kemp, "3D Wireless network localization from inconsistent distance observations," Ad Hoc & Sensor Wireless Networks, vol. 3, pp. 141–170, 2007.
- [4] R. Rubinstein and D. Kroese, "The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning," Information Science and Statistics, Springer, 2004.