

Observation Likelihood Model Design and Failure Recovery Scheme toward Reliable Localization of Mobile Robots

Chang-bae Moon¹, Woojin Chung² and Nakju Lett Doh³

^{1,2}School of Mechanical Engineering, Korea University, Seoul, KOREA

³School of Electrical Engineering, Korea University, Seoul, KOREA

Corresponding author E-mail: smartrobot@korea.ac.kr

Abstract: Although there have been many researches on mobile robot localization, it is still difficult to obtain reliable localization performance in a human co-existing real environment. Reliability of localization is highly dependent upon developer's experiences because uncertainty is caused by a variety of reasons. We have developed a range sensor based integrated localization scheme for various indoor service robots. Through the experience, we found out that there are several significant experimental issues. In this paper, we provide useful solutions for following questions which are frequently faced with in practical applications: 1) How to design an observation likelihood model? 2) How to detect the localization failure? 3) How to recover from the localization failure?

We present design guidelines of observation likelihood model. Localization failure detection and recovery schemes are presented by focusing on abrupt wheel slippage. Experiments were carried out in a typical office building environment. The proposed scheme to identify the localizer status is useful in practical environments. Moreover, the semi-global localization is a computationally efficient recovery scheme from localization failure. The results of experiments and analysis clearly present the usefulness of proposed solutions.

Keywords: Mobile robot localization, Monte Carlo Localization (MCL), Observation likelihood model, Wheel slippage, Wheeled mobile robot.

1. Introduction

There have been successful applications on autonomous navigation of mobile robots. However, developing a localization scheme for an inexperienced newcomer is still a difficult problem since there are various issues that affect to the localization results. In this paper, we concentrate on three issues which are design of observation likelihood model, localization failure detection and recovery from the failure. So far, the presented issues are difficult to be solved, and the solutions have been developed mainly based on developer's experiences.

Thrun *et.al.* introduced the Monte Carlo Localization (MCL) in (F. Dellaert, *et. al.*, 1999). The MCL technique can deal with both global localization and local localization problem. The MCL is widely used due to its superior robustness to sensor noise. Gutmann *et. al.* conducted an experimental comparison of MCL with other localization schemes in (J. S. Gutmann and D. Fox, 2002).

The observation likelihood model design has been studied extensively to solve the mobile robot self localization problem. Thrun *et.al.* presented the Beam Model in (S. Thrun, *et. al.*, 2005). The Beam Model is computed by the multiplication of individual sensor measurements. One of the disadvantages of the Beam Model is the discontinuity of the computed result. Therefore, the Beam Model requires additional filtering

schemes to exclude unmapped obstacles in a given environmental map. Another observation likelihood model is the Likelihood Field in (S. Thrun, *et. al.*, 2005). The Likelihood Field is difficult to be applied in dynamic environments because the performance becomes worse under data corruption. Therefore, it is desirable to build robust observation likelihood model under the obstacles. In addition, design requirements of observation likelihood model should be clearly presented.

Another important problem in this paper is the detection of localization failures. Sidek and Sarkar proposed the FDI framework to detect wheel slippage by using the neural network in (N. Sidek and N. Sarkar, 2006). Sidek and Sarkar exploited an additional accelerometer. Sundvall and Jensfelt proposed a method for detecting a collision due to obstacles which cannot be detected using the laser scanner in (P. Sundvall and P. Jensfelt, 2006). By using the CUSUM (F. Gustafsson, 2000) test, they estimated the difference of the expected poses between the odometry and the laser scanned image. Verma *et. al.* presented a method for detecting a fault of the Rover, which is equipped with 6 wheels, by using the particle filter (A. Doucet, *et. al.*, 2001) in (V. Verma, *et. al.*, 2004). This method cannot be applied directly to an indoor two wheeled differential mobile robot. In addition, it requires high computational cost as shown in (C. Plagemann, *et. al.*, 2007).

Observation Likelihoods	Limitations
Beam-Model (S. Thrun, <i>et. al.</i> , 2005)	- Performance drops under unmapped obstacles. Additional algorithm is required to filter out obstacles. - Probability distribution is discontinuous.
Likelihood-Field Model (S. Thrun, <i>et. al.</i> , 2005)	- No consideration of the occlusion of a line of sight during likelihood computation
Sampling Schemes	Limitations
SRL (S. Lensor, <i>et. al.</i> , 2000)	-Noise rejection and failure recovery cannot be carried out simultaneously. - Landmark based approach.
Mixture MCL (S. Thrun, <i>et. al.</i> , 2000)	- Requires additional data structure such as KD-tree to represent the proposal distribution based on the sensor measurement. - Computationally expensive because additional samples are required
Verma (V. Verma, <i>et. al.</i> , 2004)	- Computationally expensive. - Localization failure detection is not available without redundant odometry information.

Table. 1. Features and limitations of previous localization schemes.

In order to recover from the localization failure, Thrun *et. al.* introduced the MCL with the mixture proposal distribution in (S. Thrun, *et. al.*, 2000). The scheme in (S. Thrun, *et. al.*, 2000) maintains partial samples as a global localization status which is distributed based on the sensor measurements. However, it is not sufficiently fast to deal with localization failure due to wheel slippage. Lensor and Veloso introduced the Sensor Resetting Localization (SRL), which is used for soccer robot localization in (S. Lensor, *et. al.*, 2000). The SRL is difficult to apply in a large office building environment. The limitations of previous researches summarized in Table. 1. We developed a range sensor based integrated localization scheme. The proposed scheme can be used for human co-existing real environment such as a science museum or an office building. Lee and Chung introduced a localization algorithm using the selection technique that decides a strategy between the global localization and the local localization in (D. Lee and W. Chung, 2003-2006). Kim and Chung proposed a navigation behavior selection scheme that exploits localization status and navigation history in (G. Kim and W. Chung, 2007 ; W. Chung, *et. al.*, 2007) based on the Generalized Stochastic Petri-Nets (J. Wang, 1998). From our experiences, we found out several significant issues in the MCL based pose estimation. In this paper, we focus on three key issues based on the MCL scheme. The three issues are as follows:

- (1) Design requirements of a desirable observation likelihood model and performance analysis of our design.
 - (2) Localizer status estimation by computing map-matching error.
 - (3) *Semi-global localization* scheme to deal with localization failure due to the abrupt wheel slippage.
- The first issue is the design of an observation likelihood model of the MCL. The design requirements are established. Thereafter, map-matching function and probability mapping function are developed. The usefulness of the proposed design is verified through experiments.

The second issue deals with the following question: Can we rely on the current localization result? There are many exceptional situations such as severe sensor data corruption by visitors, or abrupt wheel slippage due to uneven floor conditions. In this paper, we concentrate on the localization failure due to an abrupt wheel slippage. The third issue is the *semi-global localization* scheme, which is developed to deal with the localization failure due to abrupt wheel slippage. Once the localization fails, samples should be redistributed over the entire region of the given environment to carry out a global localization. In our approach, the sample distribution area is efficiently limited to a small region by exploiting the history of the localization status.

The remainder of this paper is organized as follows. In Section 2, we present the design requirements of the observation likelihood model and our observation likelihood model. In Section 3, it is shown how to detect the localization failure. The recovery strategy is also presented. Experimental results are presented in section 4. Finally, some concluding remarks are given in section 5.

2. Observation Likelihood Model Design

2.1. Observation likelihood model design

The first step in the design of the observation likelihood model is the determination of pose error $E(\mathbf{x})$, where \mathbf{x} is a robot pose. $E(\mathbf{x})$ is determined by a sensor type, map representations, selection of features, map-matching function and so forth. $E(\mathbf{x})$ is affected by sensor noise and unmapped obstacles.

Our pose error $E(\mathbf{x})$ corresponds to the map-matching error between the scanned and predicted range images. The map-matching error should be minimized at the real pose. The error should be smoothly increased if the pose of the computed reference sample pose moves away from the real pose. This fact implies that sensitivity $\partial E(\mathbf{x})/\partial \mathbf{x}$ should not be too high.

Another significant property is that $E(\mathbf{x})$ should be insensitive to partial corruption of sensor data. Partial corruption occurs when there are unmapped obstacles around the robot and $\partial E(\mathbf{x})/\partial \mathbf{x}$ becomes smaller than the obstacle free case. However, it is desirable that $E(\mathbf{x})$ maintains its overall convergence property with reduced accuracy.

After the definition of $E(\mathbf{x})$, the next step is to design a probability mapping function $P(E)$ to perform re-sampling. The term $\partial P(\mathbf{x})/\partial \mathbf{x}$ determines the resultant convergence speed. If $\partial P(\mathbf{x})/\partial \mathbf{x}$ is too high, convergence property becomes worse. Since the pose estimation result is too much sensitive to sensor data, convergence to wrong position may take place when there exist sensor noise or data corruption due to obstacles. MCL may fail to track the robot pose because too fast convergence reduces the area of sample distribution. The larger area of sample distribution implies higher uncertainty handling capacity.

On the other hand, too small $\partial P(\mathbf{x})/\partial \mathbf{x}$ results in poor performance of pose estimation due to slow convergence. Therefore, it can be concluded that $P(E)$ and $E(\mathbf{x})$ should be carefully designed in order to achieve fast convergence and appropriate sample distribution area.

2.2. Design of map-matching error

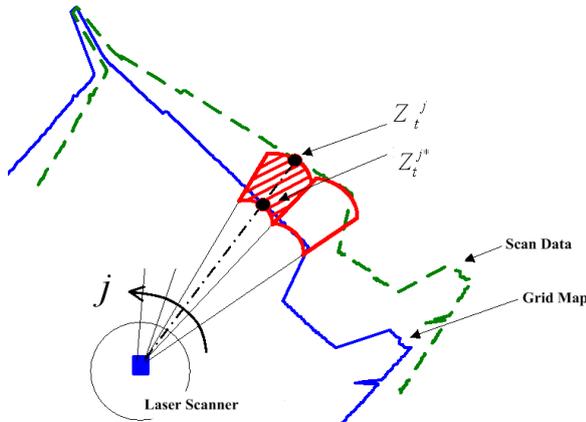


Fig. 1. Illustration of matching error.

Fig. 1 illustrates a designed map-matching error function named "matching error". The matching error $E_m(\mathbf{x})$ is defined as the difference in the area between the free space of the scanned image and the expected reference image. The expected reference distance is computed from a given grid map. The term *Scan distance* Z_t^j corresponds to j th measured distance from the scanned data. The term *Reference distance* Z_t^{j*} is a computed distance obtained by using the ray-casting algorithm. The matching error is defined by the following equations:

$$A_t^{j*} = (Z_t^{j*})^2 \Delta\theta \quad (1)$$

$$A_t^j = (Z_t^j)^2 \Delta\theta \quad (2)$$

$$\Delta A_t^j = |A_t^{j*} - A_t^j| \quad (3)$$

$$E_m(\mathbf{x}) = \left\{ \frac{\sum_j \Delta A_t^j}{\sum_j (A_t^{j*} + A_t^j)} \right\} \times 100(\%) \quad (4)$$

Equations (1) and (2) imply the area of the free spaces. Equation (4) defines the matching error $E_m(\mathbf{x})$. $\Delta\theta$ is the angular resolution of laser range finder. The denominator in eq. (4) carries out normalization. There are two roles of normalization. The first role is that the reliability of current map-matching performance can be evaluated as a bounded region from 0% to 100%. The designer can evaluate the localizer reliability by considering $E_m(\mathbf{x})$. The second role of normalization is to modulate the effect of obstacle locations. $E_m(\mathbf{x})$ becomes too large when obstacles are close to a robot without normalization.

$E_m(\mathbf{x})$ was empirically designed after testing a large variety of candidate functions. The proposed matching error shows several advantages. In the conventional approach in (S. Thrun, *et. al.*, 2005), the error is independently evaluated for each range data. The resultant positional probability is computed by multiplication of individual probabilities. Therefore, corrupted sensor data should be filtered out by using additional filtering techniques, as shown in (D. Fox, *et. al.*, 1998). However, the proposed matching error in this paper is less sensitive to partially corrupted sensor data due to obstacles.

Fig. 2 shows the result of the matching error distribution simulation when the real robot pose is located at (101,101) in the given map. For simplicity, the environment is a square and we use a grid map. The boundary of a rectangle is occupied and represented by black dots. Fig. 2(b) shows the matching error distribution with four obstacles, which are denoted by black dots. From Fig. 2(a), it is verified that *matching error* $E_m(\mathbf{x})$ is minimum at the real pose and the error surface smoothly increases according to the pose error. Fig. 2(b) shows that the overall shape of $E_m(\mathbf{x})$ surface does not change a lot even though the accuracy was decreased due to data corruption by obstacles.

Fig. 3 shows the matching error under orientation errors. It is verified that $E_m(\mathbf{x})$ is designed to achieve desirable convergence properties for all cases. Therefore, it can be concluded that the proposed map-matching function $E_m(\mathbf{x})$ satisfies the design requirements in section 2-1.

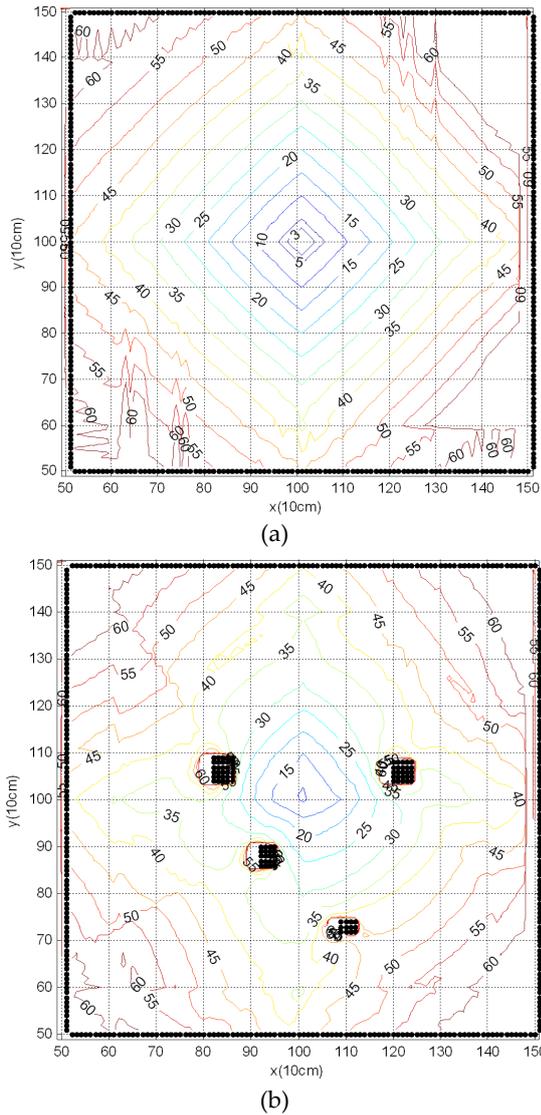


Fig. 2. The matching error distribution plot as the robot position changes for two cases: (a) without obstacle and (b) with for obstacles (marked by black dots).

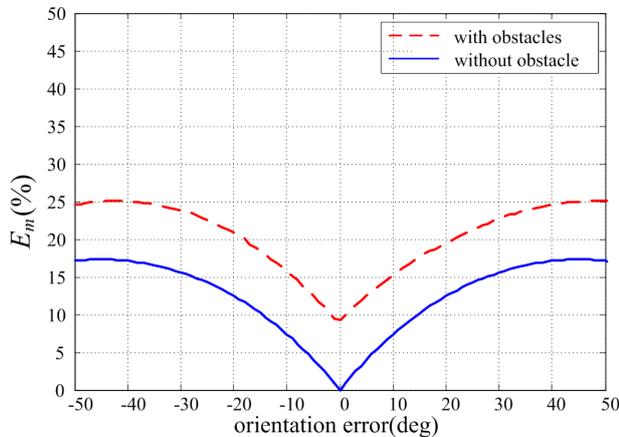


Fig. 3. The matching error plot along the changes of the orientation error.

2.3. Probability density function using the matching error

Once map-matching function is defined, a developer should design the probability update rule, which is represented by $p(o_t | \mathbf{x}_t, m) \leftarrow E_m(\mathbf{x})$. One example is to use the reciprocal error function in (D. Lee and W. Chung, 2006). However, it is difficult to estimate resultant performances of the localizer by using the reciprocal function.

Our suggestion is the use of a saturated Gaussian function as the probability density function. By monitoring the standard deviation of matching error, a designer can obtain quantitative insights into the convergence properties. Furthermore, excessive convergence can be prevented by using the saturation. The probability saturation region is designed in order to avoid erroneous concentration of probability. The localization accuracy cannot be higher than the resolution of a grid map and sensing accuracy. Therefore, it is advantageous to saturate the probability around the center of Gaussian distribution. The excessive number of iteration may reduce the pose distribution to a very small area and may disturb the uncertainty handling capability. Pfaff *et.al.* proposed a localization algorithm using an adaptive likelihood model in (P. Pfaff, *et. al.*, 2006). The adaptive likelihood model is advantageous in dynamic environments. In this paper, we exploit this algorithm. We use the statistical data of the samples, which are the mean value and standard deviation of the sample's matching error, to adjust the convergence speed. The designed probability updating rule is given by the following equations:

$$p(E_m(\mathbf{x})) = \frac{\eta}{\sqrt{2\pi\sigma_t^2}}; E_m(\mathbf{x}) \leq bias \quad (5)$$

$$p(E_m(\mathbf{x})) = \frac{\eta}{\sqrt{2\pi\sigma_t^2}} \exp\left(-\frac{E_m(\mathbf{x}) - bias)^2}{2\sigma_t^2}\right); \quad (6)$$

$$bias < E_m(\mathbf{x}) \leq \mu_t + 3\sigma_t$$

$$p(E_m(\mathbf{x})) = 0; E_m(\mathbf{x}) > \mu_t + 3\sigma_t \quad (7)$$

We compute the probability using (5), (6), and (7). η is a normalizer. The σ_t and μ_t denote for the standard deviation and mean value of sample's the matching error at time t . $Bias$ determines the region of probability saturation. The adopted $Bias$ is 3% because the grid resolution is 10cm. The region of saturation is inside of 3% $E_m(\mathbf{x})$ contour in Fig. 2(a). From Fig. 2(a), it is clear that 3% contour region is much smaller than the grid resolution. This value implies that the saturation region is about 30cm, as shown in Fig. 2(a). Equation (5) implies the saturation region of probability. The $bias$ means that if the matching error is smaller than the $bias$, the probability is saturated. Equation (7) shows if the matching error $E_m(\mathbf{x})$ is larger than $\mu_t + 3\sigma_t$. This fact implies that the region of convergence covers 99.7% of the entire probability

distribution region. Equation (6) shows the Gaussian probability function. The *bias* implies the convergence limit of the observation likelihood model.

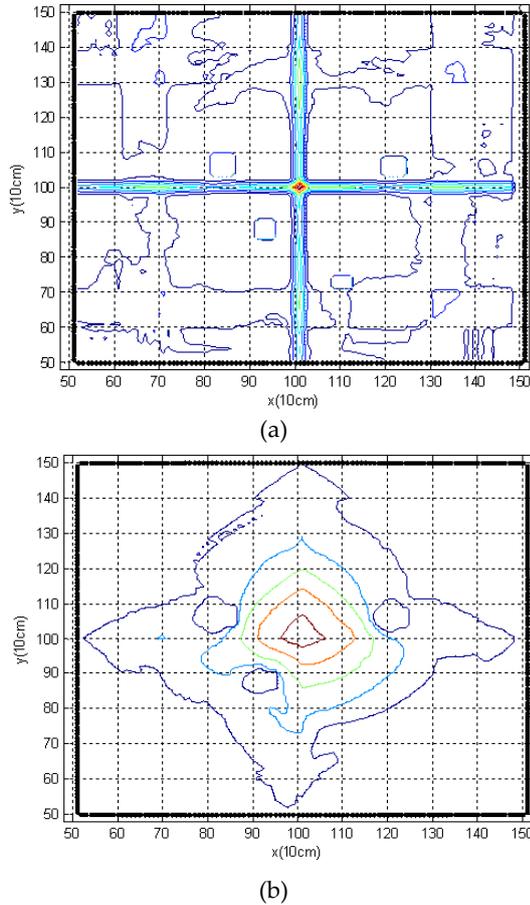


Fig. 4. The probability distribution simulation results as the robot position changes. (Red curve indicates the highest probability contour.) (a) Beam Model, (b) Proposed model.

Fig. 4 shows the probability distribution of the Beam-model and the proposed model under the identical environmental conditions with Fig. 2(b). It is well-known that the continuous distribution is more robust than the discontinuous distribution. As shown in Fig. 4(a), the probability distribution shows local discontinuity around the true position. The positional error is larger than 20cm. This result implies that the Beam-model requires a large number of samples to deal with sensor noise. However, the probability distribution of the proposed scheme is smooth around the true position as shown in Fig. 2(b). Therefore, it is advantageous to adopt the proposed observation likelihood model.

Since the proposed scheme is a range sensor based localization scheme, it works only when sensor readings are partially corrupted by obstacles. In the highly dynamic environment, additional sensors such as upward pointing cameras should be installed.

3. Localizer Status Estimation and Semi-Global Localization

3.1. Localizer status estimation using the matching error

The objective of the localizer status estimation is to decide whether the current localization status is *localizer success* or *localizer failure*. The matching error was used to detect a *localizer failure*. It is assumed that matching error is greater than a specific threshold when localization is failed. The major difficulty lies in the determination of the threshold. The threshold cannot be a single constant.

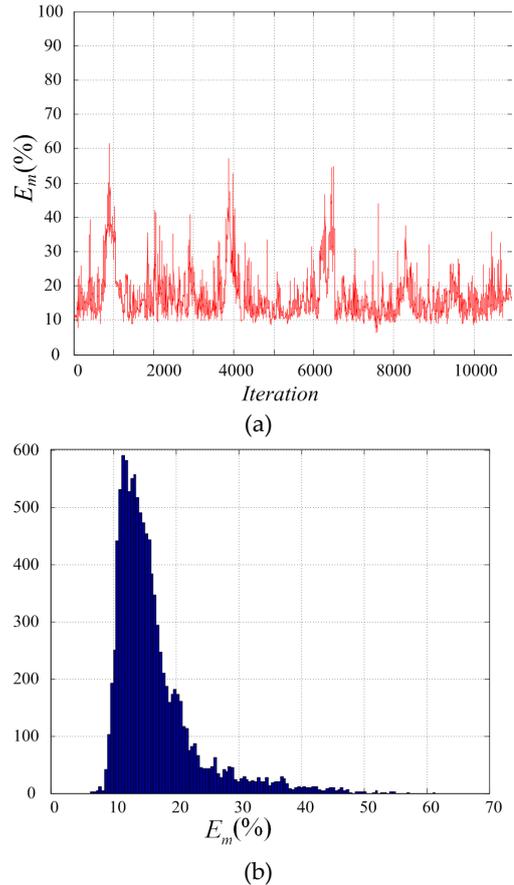


Fig. 5. Experimentally measured matching error during a navigation. (a) Matching error history, (b) Matching error histogram.

Fig. 5 shows experimentally measured the matching error history while the mobile robot performs 11,000 steps of the MCL successively. The navigation experiment is conducted in a typical office building which is shown in Fig. 7 and travel distance is about 670m. As shown in Fig. 5, the matching error is non-zero due to various errors, even for a successful localization. The matching error is affected by the scale of environment, unmapped obstacles, sensing accuracy and so forth. Therefore, it is difficult to estimate the localizer status with a constant threshold of the matching error. In this paper, the threshold h is the upper limit of the matching error of a *localizer success*.

$$h = \mu_{i-k:i} + 3\sigma_{i-k:i} + bias \quad (8)$$

Equation (8) shows the computation of threshold h . $\mu_{i-k:i}$ is the mean value of the matching error from $(i-k)$ th step to i -th step. $\sigma_{i-k:i}$ is the standard deviation of the matching error.

Equation (8) exploits the sliding window scheme based on Hypothesis testing using Gaussian and χ^2 test in (F. Gustafsson, 2000). The proposed equation is a simplified representation of the Hypothesis testing. Since bias indicates the upper bound of the Gaussian probability, the threshold is shifted by bias. The confidence interval is determined by 3σ , since the matching error does not exceed the confidence level in most cases without wheel slippage. Finally, the sliding window size k is empirically determined from experiments. k is 500, which corresponds to 100s.

The localizer status estimation is independently carried out with the MCL loop for updating cycle of sensor data, since the MCL cycle time is more slower than the sensor cycle time. The matching error can be higher than h for a short time period. In most cases, instantaneous failures can be recovered by conventional MCL owing to its robustness. Hence, we use a median filter (E. Olson, *et. al.*, 2005) to eliminate the short-term outlier values. A median filter removes the outlier by replacing a data with the median value of the n surrounding points. In the experiment, n was set to 15. The localization status is switched to *localizer failure* when the computed value of the matching error by the median filter is greater than h .

3.2. Semi-Global Localization

We propose a *semi-global localization* scheme to deal with exceptional localization situations. Many uneven floors exist in indoor environments. In such a case, abrupt wheel slippage may occur, which cannot be compensated by conventional MCL techniques. Using the localization status history of past movements, we can detect the moment that robot fell into *localizer failure*. Therefore, we can specify the reachable region of the robot. *Semi-global localization* implies that the sample distribution can be limited to the robot's reachable region.

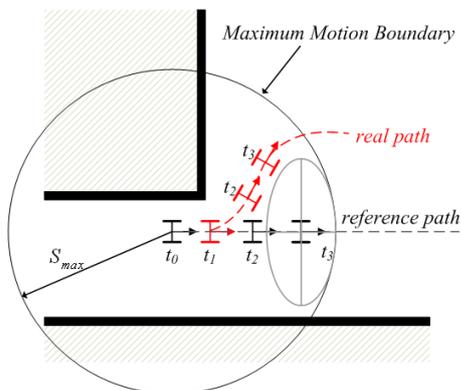


Fig. 6. Conceptual illustration of *semi-global localization*.

Fig. 6 shows the *maximum motion boundary* region as a circle after the localizer fails to estimate a robot pose. Suppose that a robot starts navigation at time t_0 following the reference path. The localizer status is estimated as a localizer success at time t_0 . While the robot moves forward, the robot's wheel slips at time t_1 . In Fig. 6, slip distance of the left wheel was larger than that of the right wheel. Since it takes time for a robot to detect *localization failure*, the robot's localization status was switched into *localization failure* at t_3 . It is clear that the local tracking may fail at t_3 , as shown in Fig. 6. The *maximum motion boundary* region of a robot after slippage is a circle, as represented in Fig. 6. The *maximum motion boundary* region is computed by assuming that the maximum moving distance is the accumulated distance from time t_0 to t_3 in all directions.

$$S_{max} = \sum_i (1 + 3\sigma) |\Delta s_i| \quad (9)$$

Equation (9) shows S_{max} , which is the maximum moving distance. σ is the standard deviation of the motion model. Δs_i is the moving distance from wheel encoders. Using a motion model that has the Gaussian probability distribution, we assume the maximum moving distance by abrupt wheel slippage as $\max(\delta_{trans}) \approx 3\sigma$. We perform the *semi-global localization* in the *maximum motion boundary* area with intentionally added 300 samples.

4. Experimental Results

4.1. Experimental environments

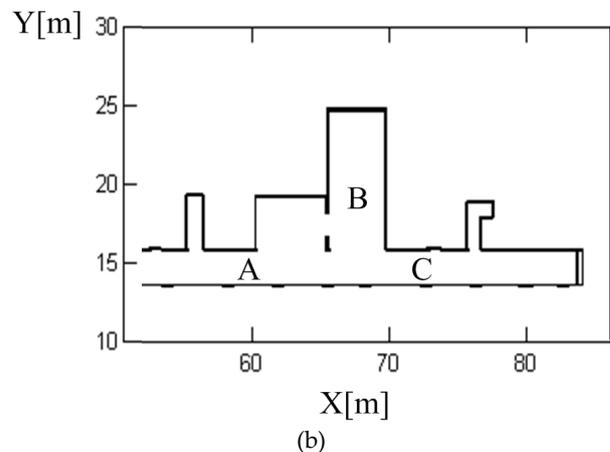
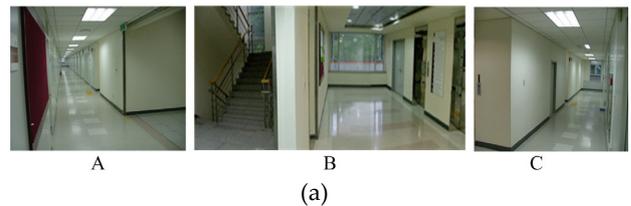


Fig. 7. Experimental environments and its grid map. (a) Target workspace, (b) Grid map.



Fig. 8. A mobile robot used for experiments.

The experiments with the proposed observation likelihood model are conducted in a large building, which is shown in Fig. 7. Fig. 7(a) shows the target workspace. The environment contains some difficulties: a mirror-like surface around the elevator hall, which is located around $x=64m\sim 70m$ and a stair, which is an open space. Fig. 7(b) shows the environment as a grid map with a grid resolution $10cm$. Fig. 8 shows a two wheeled differential drive mobile robot. The robot is equipped with two SICK laser scanners in the front and rear. The laser range finder scans 180° with 1° angular resolution and the maximum range was set to $9m$.

4.2. Observation likelihood model

Fig. 9 shows a localization result using the proposed matching error. Fig. 9 shows contour of the matching error on the left side and probability distribution on the right side. Fig. 9(a) shows first sampling result in which the samples are randomly distributed. After the first update, the samples are converged to a region in which the matching error is smaller than 60% as shown in Fig. 8(b). After 3 steps of sensor update, the samples converge to the 10% region as shown in Fig. 8(d). The experiment showed that the samples converged to the real robot pose after iterations. The samples did not converge excessively

when the saturation was used. In addition, a designer can monitor the localization result intuitively using the matching error. For example, the map-matching result can be clearly monitored quantitatively by drawing the matching error contour.

Fig. 10 shows the matching error contour under the different corruption ratios due to obstacles. A corruption ratio is a percent of corrupted sensor readings, where the range error is greater than $1m$. As shown in Fig. 10, the contour shape does not change greatly even when obstacles block the sensor. Therefore, the convergence property is still maintained, even though absolute accuracy was decreased.

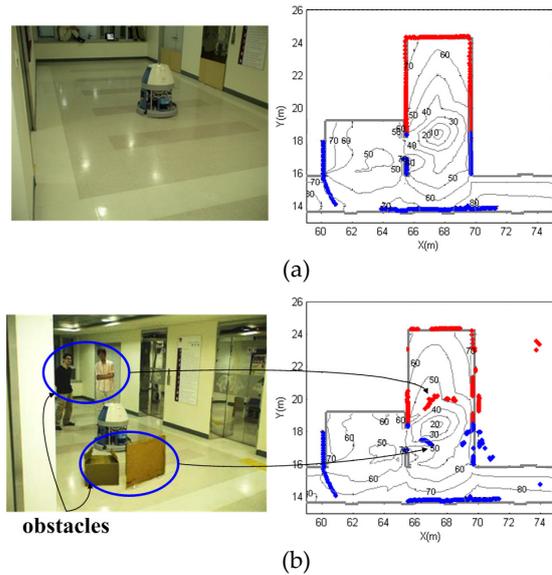


Fig. 10. Matching error contour from corruption of sensor data by obstacles. (a) Corruption ratio 0%, (b) Corruption ratio 30%.

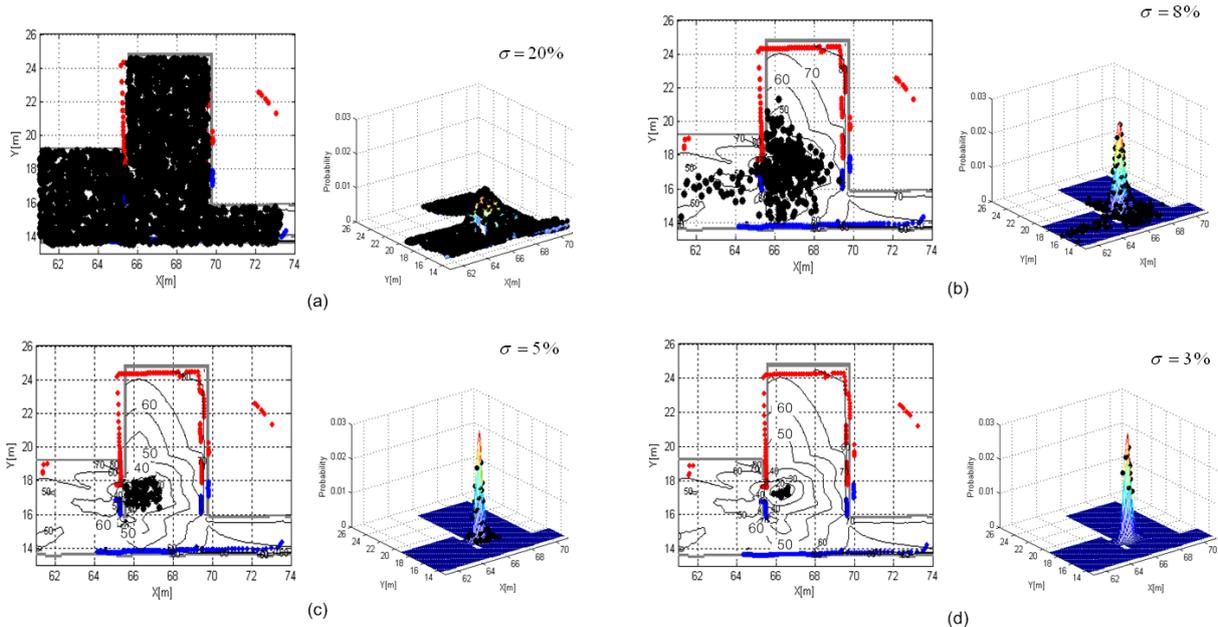


Fig. 9. Convergence of samples by MCL. (a) Initial sampling, (b) Sample distribution after 1st sensor data update, (c) Sample distribution after 2nd sensor data update, (d) Sample distribution after 3rd sensor data updates.

4.3. Localizer status estimation

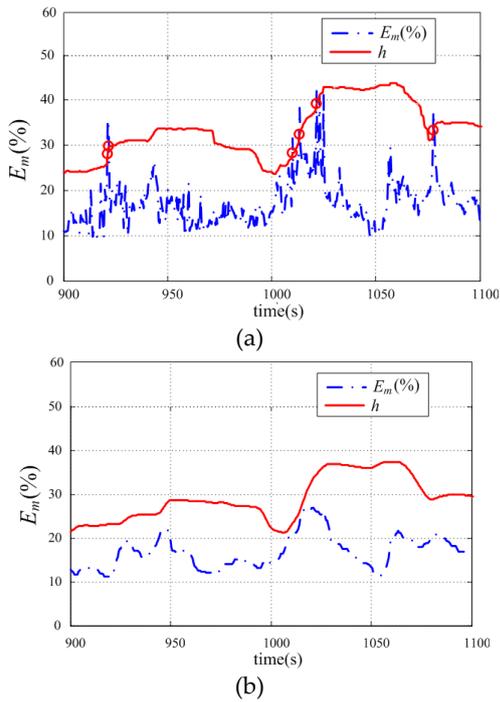


Fig. 11. Matching error history during a navigation. (a) without median filter, (b) with median filter.

Fig. 11(a) shows the measured matching error that was monitored when the robot moved in the experimental environment for about 200s. A navigation experiment was conducted in the environment shown in Fig. 7 and the localizer successfully tracked the robot pose. This means that the localization status was *localizer success* during the experiment.

Even though the status of the localizer is *localizer success*, the matching error is non-zero due to various error sources. Fig. 11(a) shows the matching error which exceeds the threshold h instantaneously, and those instants are denoted by red-circle. False alarms occurred 6 times. This result implies that the localizer estimation result might return a *localizer failure*, which is a wrong estimation result without the median filter. Fig. 11(b) shows the matching error with the median filter. As shown in Fig. 11(b), outlier values are filtered out using the median filter. Therefore, the result of the localizer status estimation was always successful without any false alarms.



Fig. 12. Wheel slippage situation due to uneven floor condition.

As shown in Fig. 12, there are two different floors in our experimental environment: one is cased with marble and the other one is cased with rubber tile. Sometimes, two wheels of the robot are stuck at the boundary of two different surfaces. As a result, the robot cannot move forward, even though the control input is pure translational motion.

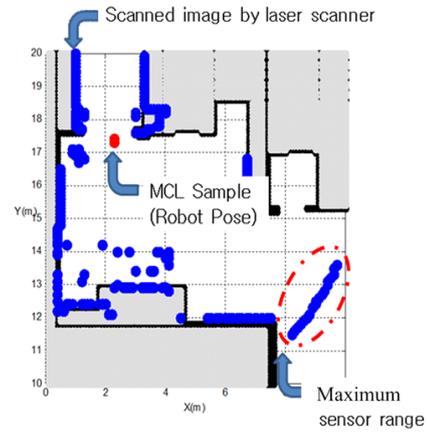


Fig. 13. Grid map and scanned image obtained with laser scanner.

Fig. 13 shows grid map and scanned image before the active wheels are stuck over the floor. As shown in Fig. 13, the scanned image (denoted by blue dots) and the grid map (denoted by black dots) matches sufficiently. This fact implies that the localizer successfully tracked the robot's pose.

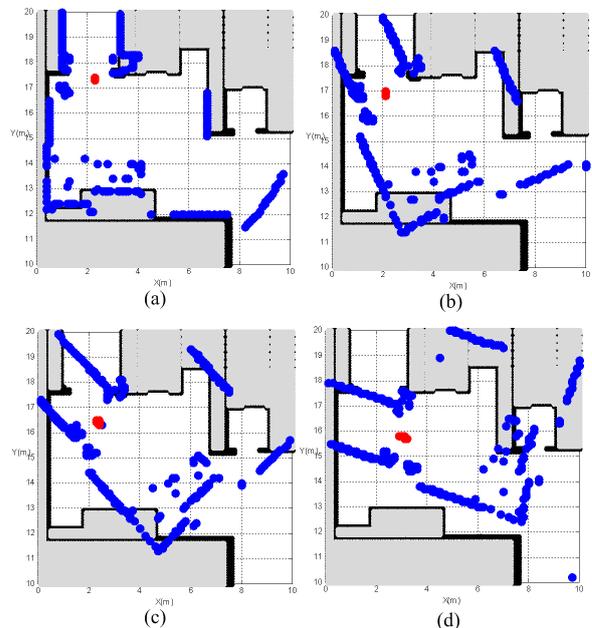


Fig. 14. Localization failure.

Fig. 14 shows *localization failure* case by abrupt wheel slippage. Even though the mobile robot cannot move to forward due to the wheel slippage, the MCL samples move forward from the real robot pose by the motion

model. As a result, the localizer failed to track the robot pose because there were no samples anymore at the real robot pose.

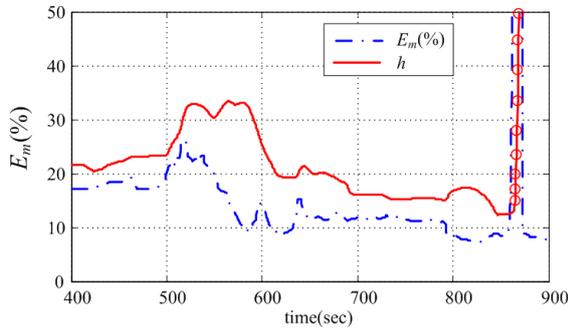


Fig. 15. Matching error history at localization failure 870s.

Fig. 15 shows the matching error history. Localizer failed to track the robot pose at 870s due to abrupt wheel slippage. As shown in Fig. 15, the matching error exceeds threshold h , which is represented as red circles. As a result of the estimation, localization status is switched to *localizer failure*. The *semi-global localization* is carried out after 870s.

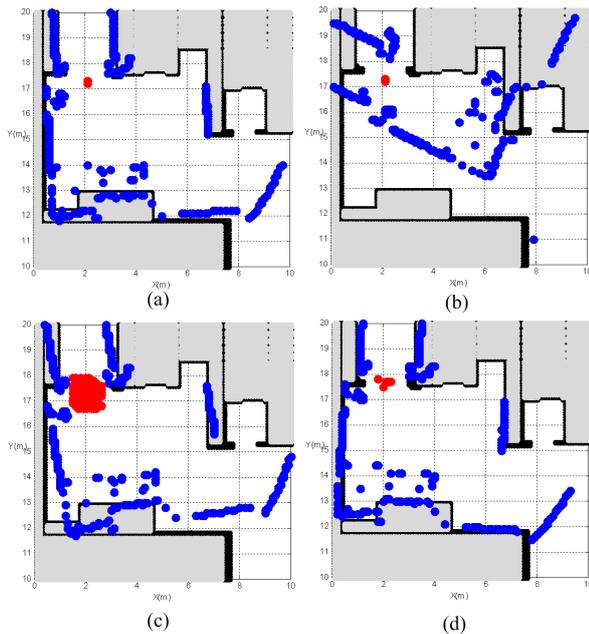


Fig. 16. *Semi-global localization*.

Fig. 16(c) shows that the localizer starts the *semi-global localization* because the matching error exceeded threshold h by the wheel slippage. As shown in Fig. 16(c), the sampling region is spread over the reachable region after slippage at (2, 17). The localizer recovers from the *localization failure* after 20s. The area of the reachable region after slippage was about $2.6m^2$. If we adopt conventional global localization instead of the proposed *semi-global localization* scheme, the samples should be spread over the whole workspace. The area of whole workspace is about $1539 m^2$. The computational cost is

proportional to the number of samples. If the uniform sample density is assumed, the number of samples is proportional to the area of sample distribution. The required sample size for the successfully global localization in the given environments is about 5,000 ~ 10,000 samples. However, the samples size for the *semi-global localization* is 300 samples. It is advantageous to use the proposed *semi-global localization* because it is computationally efficient than the conventional global localization.

5. Conclusions

Three practical issues of a reliable localization scheme were addressed and then our solutions were presented. By using the proposed analysis and design, a practical and reliable localization solution was developed. The presented approaches were successfully experimented in real environments. Recognition of the localization status greatly improves practical performance, even though the presented matching error concept is quite simple. The contributions of this paper are as follows:

- (a) Using the proposed structural observation likelihood model design scheme, inexperienced researchers have guidelines for designing an observation likelihood model.
- (b) The proposed scheme to identify the localizer status is useful in practical environments.
- (c) The *semi-global localization* is a computationally efficient recovery scheme from *localization failure*.

6. Acknowledgment

This research was supported in part by the MKE(The Ministry of Knowledge Economy), Korea, under the National Robotics Research Center for Autonomous Navigation Technology support program supervised by the NIPA(National IT Industry Promotion Agency). This research was also supported by Basic Science Research Program and Mid-career Researcher Program through the NRF grant funded by the MEST. This research was also supported by a Korea University Grant.

7. References

- F. Dellaert, D. Fox, W. Burgard and S. Thrun. (1999). Monte Carlo Localization for Mobile Robots, Proc. IEEE International Conference on Robotics & Automation (ICRA'99), Detroit, USA, May. 1999, pp. 1322-1328.
- J. S. Gutmann and D. Fox. (2002). An Experimental Comparison of Localization Methods Continued, Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02), Lausanne, Switzerland, Sep.30-Oct.4, 2002, pp. 454-459.
- S. Thrun, D. Fox and W. Burgard. (2005). *Probabilistic Robotics*, The MIT Press, 2005, pp.153-169.

- N. Sidek and N. Sarkar. (2006). Integrating Actuator Fault and Wheel Slippage Detections within FDI Framework, Proc. 5th WSEAS International Conference on Circuits, Systems, Electronics, Control & Signal Processing, Dallas, Texas, 2006, pp. 31-36.
- P. Sundvall and P. Jensfelt. (2006). Fault detection for mobile robots using redundant positioning systems, Proc. IEEE International Conference on Robotics & Automation (ICRA'06), Orlando, Florida, May 2006, pp. 3781-3786.
- F. Gustafsson. (2000). *Adaptive Filtering and Change Detection*, Wiley, 2000.
- A. Doucet, N. de Freitas, and N. Gordon. (2001). *Sequential Monte Carlo Methods in Practice*, Springer-Verlag, 2001.
- V. Verma, G. Gordon, R. Simmons and S. Thrun. (2004). Particle Filters for Rover Fault Diagnosis, *IEEE Robotics and Automation Magazine*, vol. 11, no. 2, June, 2004, pp. 56-66.
- C. Plagemann, D. Fox and W. Burgard. (2007). Efficient Failure Detection on Mobile Robots Using Particle Filters with Gaussian Process Proposals, Proc. International Joint Conference on Artificial Intelligence (IJCAI), Hyderabad, India, Jan. 6-12, 2007, pp.2185-2190.
- S. Thrun, D. Fox and W. Burgard. (2000). Monte Carlo Localization with Mixture Proposal Distribution, Proc. AAAI National Conference on Artificial Intelligence (AAAI 2000), Austin, Texas, 2000, pp.839-865.
- S. Lensor and M. Veloso. (2000). Sensor resetting localization for poorly modeled mobile robots, Proc. IEEE International Conference on Robotics & Automation (ICRA'00), San Francisco, Michigan, 2000, pp. 1225-1232.
- D. Lee, W. Chung and M. Kim. (2003). A Reliable Position Estimation Method of the Service Robot by Map Matching, Proc. IEEE International Conference on Robotics & Automation (ICRA'03), Taipei, Taiwan Sept. 14-19, 2003, pp. 2830-2835.
- D. Lee, W. Chung and M. Kim. (2004). Integrated localization of the service robot PSR, Proc. IEEE International Conference on Robotics & Automation (ICRA'04), New Orleans, LA April. 26-May. 1, 2003, pp. 4017-4022.
- D. Lee and W. Chung. (2005). Dependable Localization Strategy in Dynamic Real Environments, Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05), Edmonton, Canada, Aug. 2005, pp. 860-865.
- D. Lee and W. Chung. (2006). Discrete Status based Localization for Indoor Service Robots, *IEEE Transaction on Industrial Electronics*, vol. 53, no.5, Oct. 2006, pp. 1737-1746.
- G. Kim and W. Chung. (2007). Navigation Behavior Selection Using Generalized Stochastic Petri Nets (GSPNs) for a Service Robot, *IEEE Transaction System, Man and Cybernetics. C*, vol. 37, no. 4, July 2007, pp.494-503.
- W. Chung, G. Kim, and M. Kim. (2007). Development of Multi-Functional. Indoor Service Robot PSR Systems, *Autonomous Robots*, vol. 22, no.1, Jan. 2007, pp. 1- 17.
- J. Wang. (1998). *Timed Petri Nets Theory and Application*, Kluwer Academic Publisher, 1998.
- D. Fox, W. Burgard, S. Thrun, A. B. Cremers. (1998), Position Estimation for Mobile Robots in Dynamic Environments, Proc. Fifteenth National Conference on Artificial Intelligence. (AAAI-98), Madison, Wisconsin, 1998, pp. 983-988.
- P. Pfaff, W. Burgard and D. Fox. (2006). Robust Monte-Carlo Localization Using Adaptive Likelihood Models, *Springer Tracts in Advanced Robotics*, Springer Berlin/Heidelberg, vol. 22, 2006, pp. 181-194.
- F. Gustafsson. (2000). *Adaptive Filtering and Change Detection*, Wiley, 2000.
- E. Olson, M. Walter, S. Teller and J. Leonard. (2005). Single-Cluster Spectral Graph Partitioning for Robotics Applications, Proc. Robotics Science and Systems, RSS-05, MIT, Cambridge, MA, June, 2005, pp. 25-272.