MOBILE PLATFORM SELF-LOCALIZATION IN PARTIALLY UNKNOWN DYNAMIC ENVIRONMENTS

Patrice Boucher, Sousso Kelouwani and Paul Cohen
Perception and Robotics Laboratory, Ecole Polytechnique de Montreal
2500, Chemin de Polytechnique, Montreal, Canada

Keywords: Navigation, Localization, Dynamic environments, Point-based model, Extended Kalman Filter, 2D Point matching, Registration, Robotic platform slipping, Homogeneous matrices.

Abstract: Localization methods for mobile platforms are commonly based on an observation model that matches onboard sensors measures and environmental a priori knowledge. However, their effectiveness relies on the reliability of the observation model, which is usually very sensitive to the presence of unmodelled elements in the environment. Mismatches between the navigation map, itself an imperfect representation of the environment, and actual robot's observations introduce errors that can seriously affect positioning. This article proposes a 2D point-based model for range measurements that works with a new method for 2D point matching and registration. The extended Kalman filter is used in the localization process since it is of the most efficient tool for tracking a robotic platform's configuration in real time. The method minimizes the impact of measurement noise, mismodelling and skidding on the matching procedure and allows the extended Kalman filter observation model to be robust against skidding and unmodelled obstacles. Its $O(n \cdot m)$ complexity enables real-time optimal points matching. Simulation and experiments demonstrate the effectiveness and robustness of the proposed algorithm in dynamic and partially unknown environments.

1 INTRODUCTION

In the context of map-based navigation, a robotic platform must regularly and reliably estimate its configuration (position and orientation) within a known map of the environment. This problem is commonly referred to as the localization problem. By knowing its configuration and perceiving obstacles in the environment, the platform can choose appropriate actions in order to reach a given destination. However, moving efficiently requires an accurate localization method combined with fast real-time sensory data processing. The proposed algorithm in this paper fulfills these two requirements, using extended Kalman filtering with a novel observation model for platform localization.

Proposed by Stanley F. Schmidt in 1970 (Schmidt, 1970), the extended Kalman filter is commonly used for parameters estimation with non linear models subject to noise. The Kalman filter computes a configuration estimate in two steps. The first one is the prediction step, based on the dynamic model of the system. The second step, known as the correction step, is based on an observation model that draws relationships between the platform configuration and key measurements. In indoor environments, the localization of a robotic platform is often based on observations provided by on-board sensors such as laser range finder (Carlson et al., 2008), infrared sensor (Wei et al., 2005) and sonar sensors. Observed features are matched with a priori data about the environment in order to estimate the most plausible platform's configuration.

As explained in (Thrun et al., 2005) and (De Laet et al., 2008), the matching process from range finder data can be addressed with beams-based models, feature-based methods and correlation-based approaches. In order to deal with unmodelled objects, the beams-based and correlation-based approaches compute complex probabilistic functions given the a priori knowledge about the navigation environment (De Laet et al., 2008). Since each range measurement is considered separately, such models do not take advantage of the natural features of the surrounding platform area. On the other hand, despite that feature-based models can be robust against unexpected objects through selectivity, feature extraction and recog-

Boucher P., Kelouwani S. and Cohen P.
MOBILE PLATFORM SELF-LOCALIZATION IN PARTIALLY UNKNOWN DYNAMIC ENVIRONMENTS.
DOI: 10.5220/0002162001130120
In *Proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2009)*, page
ISBN: 978-989-674-000-9

nition may be computationally expensive and the features must be sufficiently distinctive and numerous. The lack of robustness of observation models against unmodelled objects is usually compensated by adding such objects onto the map through simultaneous localization and map building (SLAM). However, it remains attractive to have at the base a robust observation model without map modification for avoiding complications at upper levels.

For these reasons, we introduce in this paper a 2D point-based approach which works with a local occupancy grid-map instead of using direct sensor measurements or high level features. In this way, the association process is made between a set of points, extracted from the measurements, with a second set extracted from the grid-map. The configuration is then deduced by matching both sets. Two-dimensional point matching involves two main issues : pairing two sets of 2D points and geometrical matching. The most commonly used methods for geometric matching include SVD (Singular Value Decomposition) (Arun et al., 1987), unit quaternions methods (Horn, 1987) and double quaternions methods (Walker et al., 1991). Various approaches also solve the problem of pairing and matching simultaneously. Many of them are based on iterative algorithms as in (Zhang, 1994) and (Ho et al., 2007). Moreover, (Censi et al., 2005) proposes a Hough Scan Matching (HSM) approach based on the Hough Transform. However, these approaches do not explicitly mention the matching error in the mathematical formulation, a fact that cause ambiguity in the accurate evaluation of the homogeneous matrices. Since the approach presented in this paper needs robustness against matching errors caused by unmodelled objects, these methods are not convenient for a robust 2D points observation model.

In summary, the main contributions of this paper are: (1) a fast method of 2D points registration with complexity $O(n \cdot m)$ ($O(n)$ for the geometric matching) that takes into account the presence of matching errors and measurement noise for enabling realistic accuracy evaluation of the homogeneous matrices; (2) a simple and fast 2D point-based observation model that works in presence of unmodelled objects (3) a novel method for robotic platform localization based upon extended Kalman filtering. The rest of the paper is organized in five sections. Section 2 presents a mathematical formulation of the problem. In section 3, a new method for finding 2D homogeneous matrices is presented. In section 4, we present how the overall methodology can be combined with extended Kalman filtering for platform localization. Section 5 presents and discusses experimental results.

2 PROBLEM STATEMENT

The dynamic equation of a robotic platform moving in a 2D plan can be represented at each instant k by :

$$X_{k+1} = f(X_k, V_k) + \psi_k$$

where X_k is the platform state variable at instant k, V_k is the speed of the platform at instant k, ψ_k is the uncertainty (noise) on the dynamic model and $f(.,.)$ is the function used to compute the predicted state.

The observation model is represented by:

$$Z_k = h(X_k) + \xi_k$$

where Z_k represents the observations by the platform sensors, ξ_k is the uncertainty (noise) on sensor observations and $h(.)$ is the function used to get observations when the platform is in state X_k.

In real applications, f and h are non linear. In order to apply Kalman filtering, the Jacobean of f and h are computed over a nominal path. Furthermore, the following assumptions must hold:

1. ψ_k is uncorrelated with the state initial estimate;
2. ψ_k and ξ_k are uncorrelated;
3. ψ_k and ξ_k are zero mean random process.

Some of these assumptions may not hold if the following conditions occur during platform motion:

- The observations are disturbed by unmodelled obstacles;
- The platform slips on the floor.

The aim is to find an observation model that reduces significantly the impact of the platform slipping and the presence of unmodelled obstacles.

3 FINDING OPTIMAL HOMOGENEOUS TRANSFORMATION MATRICES

In this section, we present a generic method for finding homogeneous transformation matrices between two sets of 2D points.

3.1 Problem Definition

Assume 2 sets P and Q of 2D points. Assume that X_k is the state vector of the platform at time k representing its configuration in the navigation environment. P is the set of points measured by the platform sensors at configuration X_k and Q is the set of points given by

the navigation map at that configuration. P and Q are called real set and virtual set, respectively:

$$P = \{p_i, i = 1,\ldots,N\} \quad (1)$$
$$Q = \{q_i, i = 1,\ldots,N\} \quad (2)$$

We suppose that each pair $\{p_i, q_i\}$ corresponds to a single physical point in the environment. We also assume that q_i is obtained by applying the homogeneous transformation (T,R) on p_i, where T is a translation vector and R is the rotation matrix.

$$q_i = T + Rp_i \qquad \forall i \in \{1,\ldots,N\} \quad (3)$$

In the context of platform navigation with onboard sensors, the set P is affected by measurement noise. Furthermore, the real correspondence between real and virtual points is unknown. We call $\tilde{P}$ the set of noisy measurement and $\tilde{Q}$ the set of virtual points obtained from the map and affected by pairing error:

$$\tilde{P} = \{\tilde{p}_i, i = 1,\ldots,N\}$$
$$\tilde{Q} = \{\tilde{q}_i, i = 1,\ldots,N\}$$

Representing by δ_i^M the measurement error on p_i and by δ_i^C the pairing error affecting $\{p_i, q_i\}$, the following expressions can be written :

$$\tilde{p}_i = p_i + \delta_i^M \qquad \forall i \in \{1,\ldots,N\} \quad (4)$$
$$\tilde{q}_i = q_i + \delta_i^C \qquad \forall i \in \{1,\ldots,N\} \quad (5)$$

Pluggins these equations back into equation (3) yields to the expression of the pairing error δ_i^C :

$$\delta_i^C = -T - R\left(\tilde{p}_i - \delta_i^M\right) + \tilde{q}_i \qquad \forall i \in \{1,\ldots,N\} \quad (6)$$

3.2 Computing the Homogeneous Transformation Matrices

The following assumptions are made:

1. The measurement noise and pairing error are gaussian processes with zero mean and variance σ_M^2 and σ_C^2 respectively:
$$\delta_i^M \longrightarrow N(0,\sigma_M^2)$$
$$\delta_i^C \longrightarrow N(0,\sigma_C^2)$$
2. No $\{R,T\}$ other than $\{R^*,T^*\}$ minimizes the quadratic paring error.
3. The expectation of a random variable tends to be equal to its sampling average:
$$\overline{\chi_i} = E[\chi_i] = \frac{1}{N}\sum_i^N \chi_i$$

Finding T^* as Function of R^*

Inserting expressions (4) and (5) in (3), the translation vector $\tilde{t}_i$ is given by:

$$\tilde{t}_i = T^* + \delta_i^C - R^*\delta_i^M \quad (7)$$
$$= \tilde{q}_i - R^*\tilde{p}_i \quad (8)$$

and its expectation is :

$$E[\tilde{t}_i] = T^* = \overline{\tilde{q}_i} - R^*\overline{\tilde{p}_i} \quad (9)$$

Finding R^*

From equation (7), the expression δ_i^C is computed as a function of T^* and R^* :

$$\delta_i^C = \tilde{t}_i - T^* + R^*\delta_i^M \quad (10)$$

R^* and T^* must minimize the quadratic pairing error, therefore:

$$J^* = \min E\left[(\delta_i^C)^T\delta_i^C\right] \quad (11)$$

By putting equation (10) in (11), replacing $E[(\delta_i^M)^T\delta_i^M]$ by σ_M^2 and noticing that $E[(\delta_i^C)^T R^*\delta_i^M] = \sigma_M^2$ since δ_i^C and δ_i^M are correlated via (10), the following expression is obtained:

$$J^* = E\left[(\tilde{t}_i - T^*)^T(\tilde{t}_i - T^*)\right] - \sigma_M^2 \quad (12)$$

This result shows that the variance of the translation vector $\tilde{t}_i$, defined by ΔT^2, is equal to the sum of the minimum paring error variance and the measurement noise variance :

$$\Delta T^2 = E\left[(\tilde{t}_i - T^*)^T(\tilde{t}_i - T^*)\right]$$
$$= E\left[(\delta_i^C)^T\delta_i^C\right] + \sigma_M^2 \quad (13)$$

By plugging expressions (7) and (9) into (13), and by using the angle, ϕ^*, associated with the rotation matrix R^*, the expression of the cost function can be rewritten as :

$$J^* = \overline{\tilde{q}_i^T\tilde{q}_i} + \overline{\tilde{p}_i^T\tilde{p}_i} - \overline{\tilde{q}}_i^T\overline{\tilde{q}}_i - \overline{\tilde{p}}_i^T\overline{\tilde{p}}_i - \sigma_M^2 \quad (14)$$
$$-2\cos(\phi^*)\left(\overline{\tilde{p}_{ix}\tilde{q}_{ix}} + \overline{\tilde{p}_{iy}\tilde{q}_{iy}} - \overline{\tilde{p}}_{ix}\overline{\tilde{q}}_{ix} - \overline{\tilde{p}}_{iy}\overline{\tilde{q}}_{iy}\right)$$
$$-2\sin(\phi^*)\left(\overline{\tilde{p}_{iy}\tilde{q}_{ix}} - \overline{\tilde{p}_{ix}\tilde{q}_{iy}} - \overline{\tilde{p}}_{ix}\overline{\tilde{q}}_{iy} + \overline{\tilde{p}}_{iy}\overline{\tilde{q}}_{ix}\right)$$

where $\tilde{p}_i = [\tilde{p}_{ix} \quad \tilde{p}_{iy}]^T$ and $\tilde{q}_i = [\tilde{q}_{ix} \quad \tilde{q}_{iy}]^T$. Taking the derivative to be equal to zero, we obtain:

$$\phi^* = atan2\left(\overline{\tilde{p}_{ix}\tilde{q}_{iy}} - \overline{\tilde{p}_{iy}\tilde{q}_{ix}} - \overline{\tilde{p}}_{ix}\overline{\tilde{q}}_{iy} + \overline{\tilde{p}}_{iy}\overline{\tilde{q}}_{ix},\right.$$
$$\left.\overline{\tilde{p}_{ix}\tilde{q}_{ix}} + \overline{\tilde{p}_{iy}\tilde{q}_{iy}} - \overline{\tilde{p}}_{ix}\overline{\tilde{q}}_{ix} - \overline{\tilde{p}}_{iy}\overline{\tilde{q}}_{iy}\right) \quad (15)$$

The optimal rotation matrix can be deduced directly from this expression. Knowing R^*, we can find T^* by using equation (9).

Algorithm

1. Compute :

$$\overline{\tilde{p}_{ix}\tilde{q}_{iy}},\quad \overline{\tilde{p}_{iy}\tilde{q}_{ix}},\quad \overline{\tilde{p}_{ix}\tilde{q}_{ix}},\quad \overline{\tilde{p}_{iy}\tilde{q}_{iy}},$$
$$\overline{\tilde{p}_{ix}},\quad \overline{\tilde{p}_{iy}},\quad \overline{\tilde{q}_{ix}},\quad \overline{\tilde{q}_{iy}}$$

2. Find the angle ϕ^* by using equation (15) ;
3. Compute the rotation matrix $R^*(\phi^*)$;
4. Compute the translation vector T^*(9).

3.3 Evaluation of the Algorithm Complexity

The first step of the algorithm is related to the computation of averages. These operations have a complexity O(n). Steps 2, 3 and 4 are not dependent upon data size. Hence, the overall complexity is O(n).

3.4 Quality of the Homogeneous Transformation Estimation

In order to assess the quality of the result, the accuracy of the optimal translation T^* and angular error ϕ^* must be estimated for ΔT and $\Delta\phi^*$. These values are indispensable for estimating the measurement noise matrix of the Kalman Filter and (if used with Correlation-based approaches) for providing a correlation measure between the both point sets, since ΔT and $\Delta\phi^*$ increase accordingly to the inconsistency between sets by taking into account the matching error expectation.

Translation Vector Covariance Matrix

The covariance of the translation vector, Ω_T^2, is defined as :

$$\Omega_T^2 = E\left[(\tilde{t}_i - T^*)(\tilde{t}_i - T^*)^T\right] \qquad (16)$$

and can be expressed as:

$$\Omega_T^2 = \begin{bmatrix} \sigma_x^2 & \sigma_x\sigma_y \\ \sigma_y\sigma_x & \sigma_y^2 \end{bmatrix} \qquad (17)$$

The covariance matrix of the translation error corresponds to the covariance of the pairing error added to the covariance of the measurement noise reoriented so as to minimize the covariance of the translation vector, as followed :

$$\Omega_T^2 = E\left[\delta_i^C(\delta_i^C)^T\right] + R^*E\left[\delta_i^M(\delta_i^M)^T\right](R^*)^T \qquad (18)$$

Orientation Estimation Quality

Unlike for the translation, there is only one optimal rotation matrix for all pairs of 2D point. Moreover, as the optimal orientation minimizes ΔT, we can use ΔT to assess the orientation accuracy with an empirical formula. Considering that the maximum error of the angle is $\pm\pi$, we have:

$$lim_{\Delta T\to 0}\Delta\phi^* = 0 \qquad (19)$$
$$lim_{\Delta T\to\infty}\Delta\phi^* = \pi \qquad (20)$$

The following empirical formula provides a reasonable estimate of the angle accuracy :

$$\Delta\phi^* = \frac{\pi\Delta T}{\Delta T + \Delta T_{hight}} \qquad (21)$$

where ΔT_{hight} is a large translation error necessarily involving bad pairing.

3.5 Robustness Improvement

The previous results are based upon the assumption that the average pairing error is null and that the paring and measurement error are gaussian process. In particular, the presence of unknown elements in the environment entails a systematic pairing error with a non-zero average. Therefore the gaussien process assumption may not hold. Nevertheless, assuming that the proportion of unknown object points is low, equation (6) gives a good approximation of the correlation error for each pair.

In this way, it is reasonable to give more weight to pairs whose pairing error expectation δ^C is lower. To do this, we compute as first step, the transformation matrices that minimize the overall expectation of pairing error by using equations (15) and (9). Then, the correlation error of each pair can be estimated through (6), which, ignoring the measurement noise, leads to:

$$\tilde{\delta}_i^c = -T^* - R^*\tilde{p}_i + \tilde{q}_i \qquad \forall i \in \{1\cdots,N\}$$

From this result, an uncertainty coefficient can be derived as:

$$(\sigma_i^c)^2 = \tilde{\delta}_i^c * \tilde{\delta}_i^c \qquad (22)$$

The weight, γ_i, of each pair can then be defined as:

$$\gamma_i(\sigma_i^c) = \exp\{-\beta(\sigma_i^c)^2\} \qquad (23)$$

where the parameter β, which can be set empirically, determines the rejection rate for erroneous pairs. With appropriate β, this expression allows good weighting distribution among pairs .

Next, R^* and T^* can be updated by minimizing the weighted paring error expectation. The solution is identical to the one presented at (3.2), except that the

following weighted averages are used at steps 1 and 2:

$$\overline{\tilde{p}_{ix}\tilde{q}_{iy}\gamma_i},\quad \overline{\tilde{p}_{iy}\tilde{q}_{ix}\gamma_i},\quad \overline{\tilde{p}_{ix}\tilde{q}_{ix}\gamma_i},\quad \overline{\tilde{p}_{iy}\tilde{q}_{iy}\gamma_i},$$
$$\overline{\tilde{p}_{ix}\gamma_i},\quad \overline{\tilde{p}_{iy}\gamma_i},\quad \overline{\tilde{q}_{ix}\gamma_i},\quad \overline{\tilde{q}_{iy}\gamma_i}$$

3.6 Algorithm Summary

From a real points set $\tilde{P} = \{\tilde{p}_i, i = 1,\ldots,N\}$ with M candidates for each real point, such as $C_i = \{c_{ij}, j = 1,\ldots,M\}$, find a set $\tilde{Q} = \{\tilde{q}_i, i = 1,\ldots,N\}$ that depicts the same physical points of the environment linked to the navigation map.

Then,

1. Choose the nearest neighbour of set P :
$$\tilde{q}_i = arg \min_{1\leq j\leq M} dist\,(\tilde{p}_i, c_{ij})$$
where *dist* is the Euclidean distance.
2. Find R^* and T^* by geometric matching (3.2).
3. Weighting each pair with (23), recalculate R^* and T^* with weighting averages (see 3.5).

Algorithm Complexity

In sub-section (3.3), we showed that geometric matching is of O(n) complexity. As each additional point involves M extra comparisons, then the complexity of *step 1* is directly proportional to the number of points. Therefore, this step is complexity $O(n \cdot m)$. *Step 3* involves one weight computation per point, so the complexity is O(n). Hence, the total method remains complexity $O(n \cdot m)$.

4 EXTENDED KALMAN FILTER FOR LOCALIZATION

In this section, we demonstrate how the proposed points matching method can be used to enhance the robustness of platform localization based on extended Kalman filter.

4.1 Platform Dynamic Model

The platform dynamic model is usually obtained by using the dead reckoning. Assume that $[x_k, y_k, \theta_k]'$, v_k and ω_k represent respectively the configuration X_k, the translation speed and the angular speed at time k. Assume that the integration constant step (which is also the sampling period) is δt. We can then write the following dynamic equations:

$$\begin{aligned} \theta_{k+1} &= \theta_k + \delta t\,\omega_k \\ x_{k+1} &= x_k + v_k \delta t \cos\theta_k \\ y_{k+1} &= y_k + v_k \delta t \sin\theta_k \end{aligned}$$

It must be noted that the dynamic equation may change according to the choice of platform model. This is not an issue of the proposed approach since this methodology focuses only on the observation model of the Kalman theory.

4.2 2D Point-based Observation Model

This simplest observation model that could be used in conjunction with Kalman filtering is represented by the following expression:

$$\begin{bmatrix} x_{k|k} \\ y_{k|k} \\ \theta_{k|k} \end{bmatrix} = \begin{bmatrix} x_{k|k-1} + T_x^* \\ y_{k|k-1} + T_y^* \\ \theta_{k|k-1} + \phi^* \end{bmatrix} \tag{24}$$

where $\{x_{k|k-1}, y_{k|k-1}, \theta_{k|k-1}\}$ is the configuration of the platform in the navigation map coordinate system, given all observations up to time $k-1$ and $\{x_{k|k}, y_{k|k}, \theta_{k|k}\}$ is the correction based on optimal homogeneous transformation parameters, $\{T_x^*, T_y^*, \phi^*\}$, obtained at time k. The measurement noise covariance matrix is obtained by using equations (17) and (21)

$$\Xi = \begin{bmatrix} \sigma_x^2 + \varepsilon & \sigma_x\sigma_y & 0 \\ \sigma_y\sigma_x & \sigma_y^2 + \varepsilon & 0 \\ 0 & 0 & \Delta\phi^{*2} + \varepsilon \end{bmatrix} \tag{25}$$

where the added ε is required in order to keep the filter stable when the value of the translation vector covariance and the uncertainty on the estimated angle are too small. If points are defined in the robot reference frame, transformation matrices to convert them into the map frame must be applied.

When the presence of a large obstacle or multiples unknown obstacles results in severe erroneous pairings between sets P and Q, the elements on the diagonal of the noise covariance matrix Ξ increase accordingly (see expression 18). This will reduce the confidence in the observation model and increase the confidence in the dynamic model. The impact on the configuration estimate will be limited provided the disturbance is not too prolonged. Otherwise, bad observations can cause, with time, the divergence of filter. The combination *large unknown obstacle - no relative motion* is the worst situation for the localization system.

Figure 1: Controlled Navigation Environment.

Figure 2: Controlled Navigation Environment with Unmodelled Obstacles.

5 EXPERIMENTAL RESULTS

5.1 Experimentation Setup

Hardware and Software

The approach described in this paper has been implemented in C++ and many simulations have been performed in Matlab and Acropolis (Zalzal, 2006). For real experiments, Acropolis and Player-Stage(Matthias Kranz and Schmidt, 2006) have been used as the robotic framework. The mobile platform hardware is an iRobot Mini ATRV with a differential driving mode and it is equiped on the front with a Sick LMS-200 laser range finder. Laser range data has been down-sampled in order to provide 18 measurements per scan.

Navigation Environment

In order to assess the proposed methodology, a navigation environment has been built. The workspace is delimited by walls as shown on Figure (1). A map of this environment is stored on the platform on-board computer. Figure (2) shows the same navigation environment with additional unmodelled obstacles.

Localization Parameter Settings

- The typical noise magnitude on the translation and angular speeds are set empirically to $0.06m/s$, and $0.06rad/s$ respectively. Note that this noise is inflated in order to reduce the negative impact of slippage on the platform dynamic model. Since this noise reflects confidence toward the dynamic system, an inflated noise magnitude value increases the confidence toward the observation model. In situations where the platform is resistant to slippage, the dynamic model is unbiased and in this case a realistic estimate of noise gives better results.
- The rejection rate β of erroneously paired points, used in equation (23), is set empirically to 0.3.
- The value of the parameter ε, used for numerical stability of equation (25), is set to 0.0001.

External Platform Localization System

In order to obtain an accurate estimate of the platform position in the navigation environment, a second laser range finder, mounted on a fix platform is used. This device detects only the top part of the robotic platform. The positions given by this range finder are used to track the actual trajectory of the platform in the map.

5.2 Simulation for Complexity Assessment

Assessment of the pairing and registration approaches has been realized using Matlab simulation. The goal of the simulation is to find the optimal homogeneous transformation matrices corresponding to two sets of 2D points.

The size of the sets is increased from 2^2 to 2^{15} and set is randomly generated. Given $T = [2 \quad 4]'$ and $\phi = 0.333$ rad, the Q set is generated by applying this transformation matrices to set P. Using the proposed approach, given only P and Q-sets, one should recover exact values of R and T.

Figure (3) plots the execution time as a function of the set size. The average error on the estimation of T^* and ϕ^* is $4.5926e^{-15}$ and $3.2937e^{-16}$ respectively. Furthermore, the figure shows that there is a linear relationship between the set size and the execution time. This result reinforces the claim that the proposed matching method is of complexity O(n).

5.3 Experimental Tests

With the same geometrical trajectory executed repeatedly, the average tracking error is $0.12m$ with a standard deviation of $0.10m$. A typical result is illustrated on Figure (4). The solid line represents the trajectory

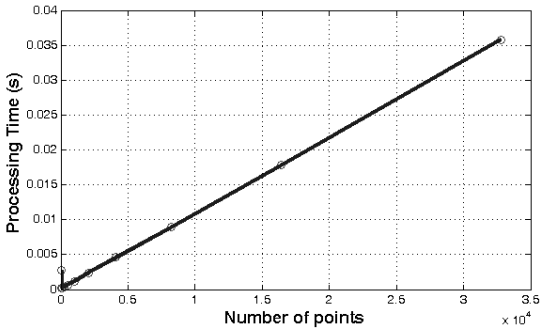

Figure 3: Processing Time as a Function of Data Size.

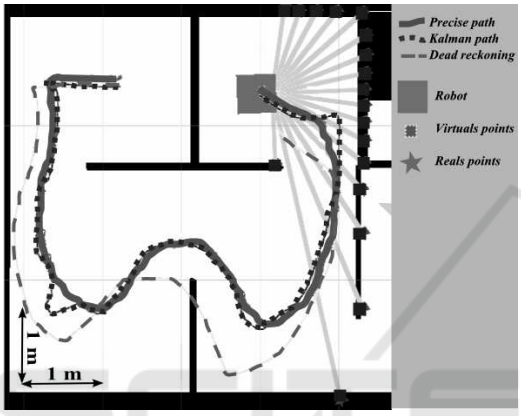

Figure 4: Configuration Tracking in The Navigation Environment.

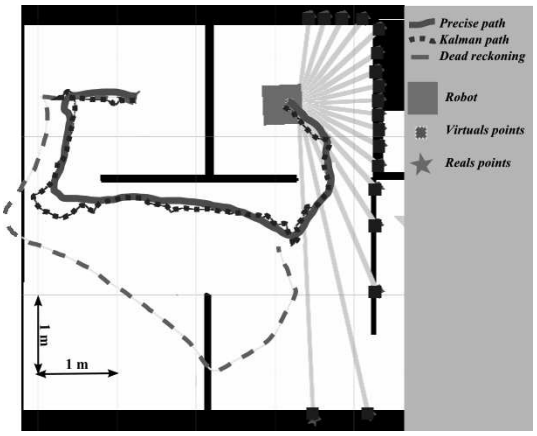

Figure 5: Configuration Tracking with Platform Slipping.

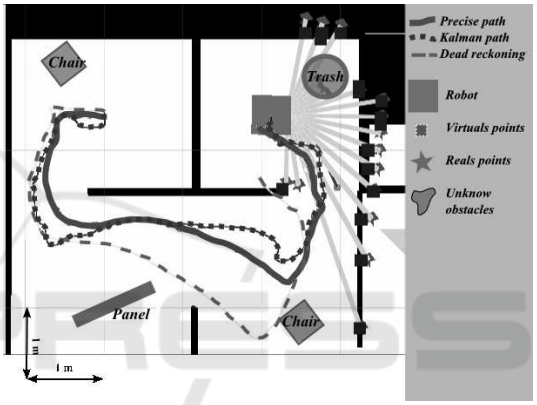

Figure 6: Configuration Tracking with Platform Slipping and Unknown Obstacles.

of the platform reported by the external laser range finder. The dotted line is the trajectory computed by using the approach described in the paper. The dashed line corresponds to navigation with dead reckoning only.

Figure (5) shows an example of severe slippage of the platform during the first quarter of the trajectory. This slippage causes an increasing deviation between the actual trajectory, as reported by the external range finder, and the estimated trajectory based upon dead reckoning. By using the proposed observation model, our estimation is similar to what has been reported by the external range finder.

For the last scenario, several unknown obstacles (unmodelled objects in the navigation map) have been added in the environment. The same geometrical trajectory has been executed repeatedly. As long as the number of observed points corresponding to unknown obstacles remains smaller that the number of points from the known environment, the estimated position of the platform as reported by the approach is still reasonably good. Figure (6) shows a successful case of path following while Figure (7) illustrates a failure of the recovery method. In order to trigger this failure situation, the robot was forced to remain stationary in front of a large unknown obstacle so that the proportion of perceived 2D points from the obstacle become preponderant for a long duration. As such contexts normally involves important mismatches between the two point sets, the measurement noise covariance matrix (equation 25) should increase accordingly, giving priority to the dynamic model. Hence, such unmodelled objects should hardly cause the divergence when the platform moves continuously. Nevertheless, despite that this paper focuses on the localization method without addressing the map building, this kind of failures could be avoided by adding such objects on the map and the general accuracy of the configuration estimate should also get increased.

6 CONCLUSIONS

This article has presented a fast 2D points matching and registration algorithm of complexity $O(n \cdot m)$ that

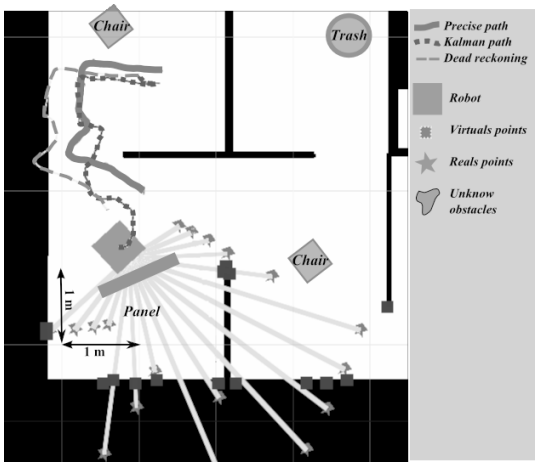

Figure 7: Configuration Tracking Failure.

exhibits robustness against erroneous point pairings. We have shown that this algorithm is easily integrable in a 2D point-based observation model for estimating a platform configuration. When used for robotic platform localization based on extended Kalman filtering, the algorithm provides an accurate estimate of the platform configuration, even in the presence of skidding and unknown obstacles in the environment. The observation model developed in this paper could be used in conjunction with other localization approaches, such as Particle Filter and Monte Carlo filtering.

ACKNOWLEDGEMENTS

This work has been supported by the Natural Science and Engineering Council of Canada (NSERC) through Grant No. CRD 349481-06. The authors wish to acknowledge the contribution of several members of the Perception and Robotics Laboratory during implementation and testing, H. Nguyen, V. Zalzal and R. Gava.

REFERENCES

Arun, K., Huang, T., and Blostein, S. (Sept. 1987). Least-squares fitting of two 3-d point sets [computer vision]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5):698 – 700.

Carlson, J., Thorpe, C., and Duke, D. L. (2008). Robust real-time local laser scanner registration with uncertainty estimation. *Springer Tracts in Advanced Robotics*, 42:349 – 357.

Censi, A., Iocchi, L., and Grisetti, G. (2005). Scan matching in the hough domain. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 2739–2744.

De Laet, T., De Schutter, J., and Bruyninckx, H. (2008). A Rigorously Bayesian Beam Model and an Adaptive Full Scan Model for Range Finders in Dynamic Environments. *Journal of Artificial Intelligence Research*, 33:179–222.

Ho, J., Yang, M.-H., Rangarajan, A., and Vemuri, B. (2007). A new affine registration algorithm for matching 2d point sets. *Proceedings - IEEE Workshop on Applications of Computer Vision, WACV 2007*.

Horn, B. (1987). Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A (Optics and Image Science)*, 4(4):629 – 42.

Matthias Kranz, Radu Bogdan Rusu, A. M. M. B. and Schmidt, A. (2006). A player/stage system for context-aware intelligent environments. *To appear in Proceedings of the System Support for Ubiquitous Computing Workshop, at the 8th Annual Conference on Ubiquitous Computing (Ubicomp 2006)*.

Schmidt, S. F. (1970). Computational techniques in kalman filtering. *NATO Advisory Group for Aerospace Research and Development*.

Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press.

Walker, M., Shao, L., and Volz, R. (1991). Estimating 3-d location parameters using dual number quaternions. *CVGIP: Image Understanding*, 54(3):358 – 67.

Wei, P., Xu, C., and Zhao, F. (2005). A method to locate the position of mobile robot using extended kalman filter. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3801 NAI:815 – 820.

Zalzal, V. (2006). *Localisation mutuelle de plates-formes robotiques mobiles par vision omnidirectionnelle et filtrage de Kalman*. PhD thesis, Ecole Polytechnique Montreal (Canada).

Zhang, Z. (1994). Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119 – 152.