Handling the Inconsistency of Relative Map Filter

Viet Nguyen, Agostino Martinelli, Roland Siegwart
Autonomous Systems Laboratory
Ecole Polytechnique Fédérale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland
Email: {*viet.nguyen, agostino.martinelli, roland.siegwart*}@*epfl.ch*

Abstract— In [5], a version of Relative Map Filter (**RMF**) is proposed to solve the simultaneous localization and map building (SLAM) problem. In the **RMF**, the map states contain only quantities invariant under shift and rotation. The estimation of the map states and their correlations is carried out in an optimal way using the Kalman filter. However, the dependency among the map states is not taken into account, thus the resulting map states are inconsistent. This paper presents two methods to enforce the consistency of the relative map states. The idea is to maintain a geometrically consistent map by solving a set of constraints between the map states. Experimental results obtained by using the proposed methods on real platform data show better performance than those deduced from the original **RMF**.

Index Terms— Localization, Mapping, SLAM, Mobile Robot Navigation.

I. Introduction

In the Simultaneous Localization and Mapping (SLAM) problem, a mobile robot has to be able to autonomously explore an unknown environment with its on-board sensors, incrementally build a map of this environment while simultaneously using this map to localize itself relative to this map.

One of the possible approaches is to use the concept of relative map. This approach has better convergence properties than, for example, the absolute map approach because the latter is based on a perfect statistical knowledge of the errors of the robot sensors (e.g. odometry) and also on the hypothesis of a linear observation, which are generally not the case for real world applications. In the relative map approach, the estimation process involves only elements which are invariant to the robot motion errors. Furthermore, the relative approach has better computational scaling properties than the absolute map approach.

Much of research has been carried out on studying the relative map approach. The first mathematical formulation was given in [8], and later [2] introduced a relative map based on quantities invariant to the robot pose, i.e. to shift and rotation. The same idea was adopted in [3]. Both algorithms estimate the relative distances between landmarks pairwise. However, it has been pointed out in [5] that the algorithms are suboptimal because they do not take into account any correlation between the distances.

Another relative map algorithm has been proposed in [7], [6] in which a relative map filter is coupled with a Geometric Projection Filter (GPF) in the estimation. The second filter provides a means to produce a geometrically consistent map from the relative map, by solving a set of linear constraints. Both filters are optimal since the dynamics/observation equations are both linear and they are based on the Kalman Filter. However, the elements used in this algorithm are invariant to shift only, not to rotation.

Recently, a new relative map filter was proposed in [5], [4]. In addition to considering only quantities invariant to shift and rotation (i.e. distances between landmarks), the algorithm also takes into account the correlation between the map states. The estimation is then carried out by applying a Kalman filter which is optimal for the modeled system of linear dynamics/observation equations. However, the estimated quantities are not independent and the algorithm has no means to enforce the dependency between the map states. Consequently, the divergence problem may arise if the dynamics/observation and error models are not perfect and, in general, it is the case. This paper presents two methods to fill the gap. The methods play a similar role as the second half of the GPF where they are to maintain the relative map geometrically consistent.

The following section gives a short review of the Relative Map Filter (RMF) presented in [5]. An inconsistency analysis of the RMF is then given. The next two sections describe two methods for consistency enforcement with their properties and experimental performance. Conclusions are presented in the last section.

II. The Relative Map Filter (RMF)

In the RMF, the map state contains only relative quantities between landmarks. In this paper, we consider only relative distances between point landmarks. Of course, the distances are quantities invariant under shift and rotation, i.e. they are independent of the robot configuration. Let's denote $\mathbf{d}$ as the state and $\mathbf{P}$ as its covariance matrix. In Fig.1a, the vector $\mathbf{d}$ contains the indicated distances between the 6 landmarks. Clearly, not all of the distances between the 6 landmarks are stored in $\mathbf{d}$ because not all the landmarks are observed together at the same time. At a given time step, the observation consists of a set of distances between the landmarks observed by the robot through its external sensor (Fig.1b). These distances may have been observed (i.e. already in $\mathbf{d}$) or may not. Let's introduce the following notations:

$$\mathbf{d}_{old} = [\mathbf{u}, \mathbf{w}_{old}]^T \qquad \mathbf{d}_{obs} = [\mathbf{w}_{obs}, \mathbf{v}]^T$$

where $\mathbf{d}_{old}$ is the state estimated at a given time step just before a new observation is made; $\mathbf{d}_{obs}$ is the observation
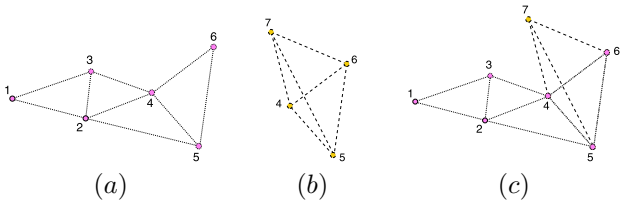
Fig. 1. (a) The relative map before the observation, (b) the observation, and (c) the relative map obtained by fusing the information coming from the old map and the observation. In all the three figures the map state only contains the indicated distances between the landmarks.

Fig. 2. The Inconsistency of the relative map estimates.

at the same time step, containing a set of distances between the landmarks observed by the robot. $\mathbf{u}$ contains the distances which are not re-observed (i.e. which do not appear in the vector $\mathbf{d}_{obs}$). $\mathbf{w}_{old}$ contains the distances re-observed (denoted by $\mathbf{w}_{obs}$ in the vector $\mathbf{d}_{obs}$). Finally, $\mathbf{v}$ contains the distances observed for the first time at the considered time step. The associated covariance matrices are:

$$\mathbf{P}_{old} = \begin{bmatrix} \mathbf{P}_{uu} & \mathbf{P}_{uw} \\ \mathbf{P}_{uw}^T & \mathbf{P}_{ww} \end{bmatrix} \qquad \mathbf{P}_{obs} = \begin{bmatrix} \mathbf{R}_{ww} & \mathbf{R}_{wv} \\ \mathbf{R}_{wv}^T & \mathbf{R}_{vv} \end{bmatrix}$$

We adopt the following notations to denote the newly estimated quantities, obtained by fusing the old state with the observed one (Fig. 1c):

$$\mathbf{d}_{new} = [\mathbf{u}_{new}, \mathbf{w}_{new}, \mathbf{v}_{new}]^T$$

$$\mathbf{P}_{new} = \begin{bmatrix} \mathbf{Pn}_{uu} & \mathbf{Pn}_{uw} & \mathbf{Pn}_{uv} \\ \mathbf{Pn}_{uw}^T & \mathbf{Pn}_{ww} & \mathbf{Pn}_{wv} \\ \mathbf{Pn}_{uv}^T & \mathbf{Pn}_{wv}^T & \mathbf{Pn}_{vv} \end{bmatrix}$$

We obtain the new estimates of the state and its covariance matrix by applying the equations of the Kalman filter. Observe that the observation is linear in the state (is the identity) and therefore the Kalman filter is optimal.

$$\begin{aligned}
\mathbf{u}_{new} &= \mathbf{u} + \mathbf{P}_{uw}(\mathbf{P}_{ww} + \mathbf{R}_{ww})^{-1}(\mathbf{w}_{obs} - \mathbf{w}_{old}) \\
\mathbf{w}_{new} &= \mathbf{w}_{old} + \mathbf{P}_{ww}(\mathbf{P}_{ww} + \mathbf{R}_{ww})^{-1}(\mathbf{w}_{obs} - \mathbf{w}_{old}) \\
\mathbf{v}_{new} &= \mathbf{v} + \mathbf{R}_{wv}^T(\mathbf{P}_{ww} + \mathbf{R}_{ww})^{-1}(\mathbf{w}_{old} - \mathbf{w}_{obs}) \\
\mathbf{Pn}_{uu} &= \mathbf{P}_{uu} - \mathbf{P}_{uw}(\mathbf{P}_{ww} + \mathbf{R}_{ww})^{-1}\mathbf{P}_{uw}^T \\
\mathbf{Pn}_{uw} &= \mathbf{P}_{uw} - \mathbf{P}_{uw}(\mathbf{P}_{ww} + \mathbf{R}_{ww})^{-1}\mathbf{P}_{ww} \\
\mathbf{Pn}_{uv} &= 0 \\
\mathbf{Pn}_{ww} &= \mathbf{P}_{ww} - \mathbf{P}_{ww}(\mathbf{P}_{ww} + \mathbf{R}_{ww})^{-1}\mathbf{P}_{ww} \\
\mathbf{Pn}_{wv} &= \mathbf{R}_{wv} - \mathbf{R}_{ww}(\mathbf{P}_{ww} + \mathbf{R}_{ww})^{-1}\mathbf{R}_{wv} \\
\mathbf{Pn}_{vv} &= \mathbf{R}_{vv} - \mathbf{R}_{wv}^T(\mathbf{P}_{ww} + \mathbf{R}_{ww})^{-1}\mathbf{R}_{wv}
\end{aligned}$$

(Hint: Apply the Kalman update equations to $[\mathbf{u}, \mathbf{w}_{old}]^T$, $[\mathbf{w}_{obs}, \mathbf{v}]^T$ with observations $[\mathbf{w}_{obs}]$, $[\mathbf{w}_{old}]$, respectively.)

The RMF's update time is dominated by the 3 updates of $\mathbf{u}_{new}, \mathbf{Pn}_{uu}, \mathbf{Pn}_{uw}$ (other vectors/matrices have constant sizes, thus constant update time). It is possible to show that the update time for $\mathbf{u}_{new}, \mathbf{Pn}_{uu}, \mathbf{Pn}_{uw}$ is also constant time. (Hint: since the covariance matrix is block diagonal, $\mathbf{Pn}_{uw}$ has constant non-zero rows. Thus, multiplications in the 3 update equations result in a matrix that contains a non-zero block with constant size.) Overall, the RMF has a constant update time.

The absolute map can be recovered from the estimated relative map by a relative transformation, given the absolute
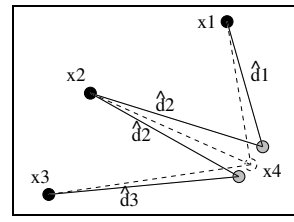
locations of two landmarks. These two seeding landmarks can be provided or taken from the first observation. At a given time step, the absolute location of a landmark can be determined from the absolute locations of two available landmarks by the following formula:

$$\begin{aligned}
x &= \frac{x_2 + x_1}{2} + \frac{(x_2 - x_1) \times (d_1^2 - d_2^2)}{2 \times r^2} \pm \frac{y_2 - y_1}{2 \times r^2} \times \\
&\quad \times \sqrt{[(d_1 + d_2)^2 - r^2] \times [r^2 - (d_2 - d_1)^2]} \\
y &= \frac{y_2 + y_1}{2} + \frac{(y_2 - y_1) \times (d_1^2 - d_2^2)}{2 \times r^2} \mp \frac{x_2 - x_1}{2 \times r^2} \times \\
&\quad \times \sqrt{[(d_1 + d_2)^2 - r^2] \times [r^2 - (d_2 - d_1)^2]}
\end{aligned}$$

or in function form:

$$\mathbf{x} = [x \quad y]^T = h(d_1, d_2) \qquad (1)$$

where $\mathbf{x}_1 = (x_1, y_1)$, $\mathbf{x}_2 = (x_2, y_2)$ are two available landmarks, r is the distance between them; d_1, d_2 are the distances between the two landmarks to the considered one. Note that the equations give two possible solutions, but an approximate location of $\mathbf{x}$ can be used to select the right solution. In (1), it uses the fact that $\mathbf{x}_1$ and $\mathbf{x}_2$ are given or already computed from previous steps, therefore x_1, y_1, x_2, y_2, r are considered fixed.

By applying recursively the method for all the landmarks, starting from the two seeding landmarks, it is possible to obtain their absolute locations at each time step. If the consistency (see next section for the definition) between the estimated distances is achieved, then clearly the choice of the two seeding landmarks will not have any influence on the result.

III. THE INCONSISTENCY OF RELATIVE MAP ESTIMATES

Under perfect conditions, e.g. the robot sensors were not affected by any error and the dynamics/observation models perfectly represented the system, the relative map obtained from the RMF would contain no inconsistency. However, it is impossible to achieve those perfect conditions in practice. Thus, transforming from an estimated relative map to an absolute map produces errors and inaccuracies, and they are dependent on the sequences of the relative transforms.

Fig.2 depicts an example of such inconsistency. The absolute location of landmark $\mathbf{x}_4$ can be recovered from the absolute locations of landmarks $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$ and the estimated distances $\hat{d}_1, \hat{d}_2, \hat{d}_3$. However, due to the imperfection discussed above, the relative transformation results in

two inconsistent solutions: one obtained from the set $\{\mathbf{x}_1, \mathbf{x}_2, \hat{d}_1, \hat{d}_2\}$ and one obtained from $\{\mathbf{x}_2, \mathbf{x}_3, \hat{d}_2, \hat{d}_3\}$ (the third solution obtained from $\{\mathbf{x}_3, \mathbf{x}_1, \hat{d}_3, \hat{d}_1\}$ is dependent and can be deduced from the other two solutions). A similar inconsistency problem is also observed and tackled in the Geometric Projection Filter. Thus, the following definition is adopted for the relative map consistency ([7]):

> *A relative map is consistent if all possible transformations to an absolute map yield unique and unambiguous absolute landmark locations.*

The statement implies that if the transformation is applied recursively from the two known landmarks (the seeding landmarks), and produces unique solution for each landmark location, then the relative map is consistent.

IV. THE RELATIVE MAP GEOMETRIC FILTER (RMGF)

The basic idea is to use the absolute landmark locations as the fusing points to enforce the consistency of the estimated relative map. In Fig.2, if we are able to fuse or unite the two inconsistent solutions of landmark $\mathbf{x}_4$, we can obtain a unique consistent solution for the absolute location of $\mathbf{x}_4$. In other words, if the consistency enforcement is applied recursively for all the landmarks, we will obtain a consistent relative map and thus be able to recover the absolute map. Note that in the original RMF, the absolute locations of landmarks are only used for data association.

The next two subsections IV-A, IV-B will describe in turn the two proposed approaches of how to interpret the "fusion" as geometric constraints and how to impose the constraints into the estimation process. In subsection IV-C, it will be shown that the two approaches have different formulations and consequently different characteristics subjected to linearization errors.

Fig.3 shows the structure of the RMGF algorithm. It consists of two main components. The first component is essentially the original RMF algorithm. The resulting relative map of this stage has no consistency enforcement, thus it is inconsistent in general. The second component of the RMGF is connected to the first half by a switch, through which the unconstrained relative map ($\hat{\mathbf{d}}_r(k|k)$ and $\hat{\mathbf{P}}_r(k|k)$) is fed into a consistency enforcement system. In this system, the geometric constraints between map states are applied by a Kalman filtering update. The result of this updating process is a consistent relative map and the process is repeated.
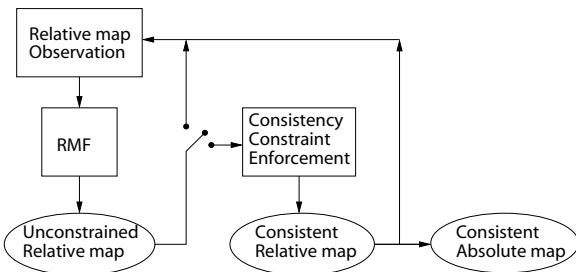
A. Standard Approach - The RMGF-SA

Fig.4 shows a possible situation of inconsistency when recovering the absolute map from an unconstrained relative map. Landmark $\mathbf{x}_5$ is connected with landmarks $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4$ by the estimated distances $\hat{d}_1, \hat{d}_2, \hat{d}_3, \hat{d}_4$, respectively. However, due to inconsistent errors, 3 possible inconsistent solutions are obtained when recovering the absolute location of $\mathbf{x}_5$ as shown (other combinations are dependent and can be deduced from these 3 solutions).

Clearly, in order to be consistent, the 3 solutions $\mathbf{x}_{12}$, $\mathbf{x}_{23}$, $\mathbf{x}_{34}$ must be united into a unique landmark $\mathbf{x}_5$. Equivalently, the following constraints must be satisfied:

$$\begin{bmatrix} \mathbf{x}_{12} - \mathbf{x}_{23} \\ \mathbf{x}_{12} - \mathbf{x}_{34} \end{bmatrix} = 0 \tag{2}$$

The fact that $\mathbf{x}_{12}$ is a function of $\mathbf{x}_1, \mathbf{x}_2, \hat{d}_1, \hat{d}_2$; and $\mathbf{x}_1, \mathbf{x}_2$ are considered fixed (given or already computed in previous steps), $\mathbf{x}_{12}$ is a function of $\hat{d}_1, \hat{d}_2$. The same argument is applied to $\mathbf{x}_{23}$ and $\mathbf{x}_{34}$. Using (1), (2), we have:

$$\begin{bmatrix} h(\hat{d}_1, \hat{d}_2) - h(\hat{d}_2, \hat{d}_3) \\ h(\hat{d}_1, \hat{d}_2) - h(\hat{d}_3, \hat{d}_4) \end{bmatrix} = 0 \tag{3}$$

or in a compact form:

$$\mathbf{H}(\hat{\mathbf{d}}_r) = 0 \tag{4}$$

If we interpret (4) as a perfect observation of $\mathbf{z} = \mathbf{H}(\hat{\mathbf{d}}_r(k|k))$ with no observation noise, applying the constraint in an Extended Kalman filter, we have:

$$\hat{\mathbf{d}}_{cr}(k|k) = \hat{\mathbf{d}}_r(k|k) + \mathbf{K}(k|k)(-\mathbf{H}(\hat{\mathbf{d}}_r(k|k))) \tag{5}$$

$$\hat{\mathbf{P}}_{cr}(k|k) = \hat{\mathbf{P}}_r(k|k) - \mathbf{K}(k|k)\nabla\mathbf{H}_k\hat{\mathbf{P}}_r(k|k) \tag{6}$$

$$\mathbf{K}(k|k) = \hat{\mathbf{P}}_r(k|k)\nabla\mathbf{H}_k^T[\nabla\mathbf{H}_k\hat{\mathbf{P}}_r(k|k)\nabla\mathbf{H}_k^T]^{-1} \tag{7}$$

where $\nabla\mathbf{H}_k$ is the Jacobian of $\mathbf{H}$ evaluated at $\hat{\mathbf{d}}_r(k|k)$; $\hat{\mathbf{d}}_{cr}(k|k)$ is the constrained relative map state, $\hat{\mathbf{P}}_{cr}(k|k)$ is the associated covariance matrix. It is easy to verify that $\hat{\mathbf{d}}_{cr}(k|k)$ satisfies the linearization of the consistency constraint (4) (pre-multiply both sides of (5) by $\nabla\mathbf{H}_k$).

This update is performed recursively at each landmark in the absolute map recovering process, starting from the seeding landmarks. The number of rows of $\nabla\mathbf{H}_k$ is the number of constraints, which equals to $p - 2$, where p is the number of given/previously computed landmarks. Also notice that there are only 3 nonzero elements in each row of $\nabla\mathbf{H}_k$ (see (3)) and since $\hat{\mathbf{P}}_r$ is block diagonal, $\hat{\mathbf{P}}_{cr}$ remains block diagonal.
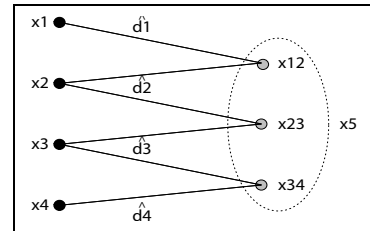

Fig. 3.   The structure of the RMGF algorithm.

Fig. 4.   The Standard Approach: At the fusing landmark $\mathbf{x}_5$, there are two constraints: $\mathbf{x}_{12}$ - $\mathbf{x}_{23} = 0$ and $\mathbf{x}_{12}$ - $\mathbf{x}_{34} = 0$.

Using the positive definite property of the matrices, it is easy to prove that applying the constraints can decrease the uncertainty in each estimated relative map covariance (using (6), (7)). As the number of observations tends to infinity, the relative map becomes perfectly known. The equations (5), (6), (7) can be performed in constant time using *Sequential Processing* since the observation has no noise (see [1]). Thus, the consistency update for all the landmarks is in linear time. Overall, the complexity of the RMGF-SA is linear in the number of landmarks.

B. Indirect Approach - The RMGF-IA

We consider again the previous example (Fig.5). Instead of writing directly the constraints as in (2), we can make the following reasoning: "fusing" $\mathbf{x}_{12}$, $\mathbf{x}_{23}$, $\mathbf{x}_{34}$ is equivalent to equalizing q_3 to $\hat{d}_3$ and q_4 to $\hat{d}_4$ (q_3, q_4 are the distances from $\mathbf{x}_3$, $\mathbf{x}_4$ to $\mathbf{x}_{12}$ as shown). Thus, if we interpret q_3, q_4 as *virtual observations* of $\hat{d}_3$, $\hat{d}_4$, then we can apply again the RMF's equations to update the map state.

The indirect approach can be followed in 3 steps:

Step 1: Select one possible solution as the pivot, for example, compute $\mathbf{x}_{12}$ and its covariance matrix $\mathbf{P}_{\mathbf{x}_{12}}$ using (1):

$$\mathbf{x}_{12} = h(\hat{d}_1, \hat{d}_2) \qquad \mathbf{P}_{\mathbf{x}_{12}} = \nabla h_k \mathbf{P}_{d_1, d_2} \nabla h_k^T \quad (8)$$

Step 2: Compute the virtual observation $[q_3 \; q_4]^T$ of $\mathbf{z} = [\hat{d}_3 \; \hat{d}_4]^T = \mathbf{C}\hat{\mathbf{d}}_r$ and its covariance matrix $\mathbf{R}$ from $\mathbf{x}_{12}$, $\mathbf{P}_{\mathbf{x}_{12}}$ using:

$$d = \|\mathbf{x}_{12} - \mathbf{x}^*\|_2 \quad \text{or} \quad d = g_{\mathbf{x}^*}(\mathbf{x}_{12}) \quad (9)$$

where d is the distance between $\mathbf{x}_{12}$ and a given/previously computed landmark $\mathbf{x}^*$ (in this case $\mathbf{x}_3$ or $\mathbf{x}_4$), and:

$$\mathbf{b} = \begin{bmatrix} q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} g_{\mathbf{x}_3}(\mathbf{x}_{12}) \\ g_{\mathbf{x}_4}(\mathbf{x}_{12}) \end{bmatrix} = \mathbf{G}(\mathbf{x}_{12}) \quad (10)$$

$$\mathbf{R} = \nabla \mathbf{G}_{\mathbf{x}_{12}} \mathbf{P}_{\mathbf{x}_{12}} \nabla \mathbf{G}_{\mathbf{x}_{12}}^T \quad (11)$$

Step 3: Apply the virtual observation $\mathbf{z} = \mathbf{C}\hat{\mathbf{d}}_r$ in a linear Kalman filter:

$$\hat{\mathbf{d}}_{cr}(k|k) = \hat{\mathbf{d}}_r(k|k) + \mathbf{K}(k|k)(\mathbf{b} - \mathbf{C}\hat{\mathbf{d}}_r(k|k)) \quad (12)$$

$$\hat{\mathbf{P}}_{cr}(k|k) = \hat{\mathbf{P}}_r(k|k) - \mathbf{K}(k|k)\mathbf{C}\hat{\mathbf{P}}_r(k|k) \quad (13)$$

$$\mathbf{K}(k|k) = \hat{\mathbf{P}}_r(k|k)\mathbf{C}^T[\mathbf{C}\hat{\mathbf{P}}_r(k|k)\mathbf{C}^T + \mathbf{R}]^{-1} \quad (14)$$

where we use the same notations as before. Note that the matrix $\mathbf{C}$ has the same number of rows as matrix $\nabla \mathbf{H}_k$, but each row of $\mathbf{C}$ has only one nonzero element which equals 1 (in the considering example, $\mathbf{C}$ has only two 1 elements, corresponding to $\hat{d}_3$ and $\hat{d}_4$).

This RMGF-IA has the same properties as for the previous approach: the uncertainty in the relative map covariance decreases; the relative map becomes perfectly known if the number of observations tends to infinity. The same decomposition technique used in the RMF can be used to decompose the estimation equations (12), (13), (14). Therefore, the consistency update for one landmark is constant time and for all the landmarks is linear time. Overall complexity of the RMGF-IA is linear in the number of landmarks.
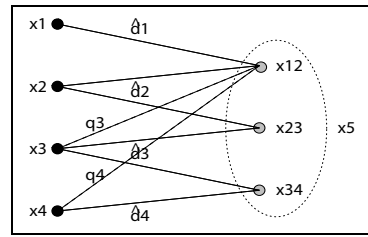

Fig. 5. The Indirect Approach: At the fusing landmark x_5, there are two constraints: $\hat{d}_3 = q_3$ and $\hat{d}_4 = q_4$.

Note that an approximation is made when introducing the *virtual observation* $\mathbf{z} = \mathbf{C}\hat{\mathbf{d}}_r$. Theoretically, a perfect observation is preferred since it contains no noise, whereas a virtual observation contains some noise which is accounted for in the associated covariance matrix $\mathbf{R}$. One way to limit the "artificial" imperfection is to select the best one (e.g. with smallest error which can be provided by $\mathbf{P}_{\mathbf{x}_{12}}$ in (8)) among the possible solutions as the pivot. In the limit, when the uncertainty of the chosen pivot approaches 1 (i.e. surely certain), the virtual observation becomes a perfect observation.

C. Linearization errors

This subsection devotes to analyze the performance of the two approaches, the estimators $\{(5)-(6)-(7)\}$ and $\{(12)-(13)-(14)\}$, subjected to linearization errors when the observation data are affected by a large noise. We consider $\hat{\mathbf{d}}_r$, $\hat{\mathbf{P}}_r$ as the input and $\hat{\mathbf{d}}_{cr}$, $\hat{\mathbf{P}}_{cr}$ as the output of the two estimators. The time step k is omitted for clarity.

The RMGF-SA uses an EKF as its estimator. In (7), the gain $\mathbf{K}$ is strongly and directly affected by the error in $\nabla \mathbf{H}_k$. Observing from (1), h is highly nonlinear. Thus, if the noise error in the input $(\hat{\mathbf{d}}_r, \hat{\mathbf{P}}_r)$ is large, then the linearization error of $\nabla \mathbf{H}_k$ evaluated at an "erroneous" $\hat{\mathbf{d}}_r$ can be very large, far from the true value. (In fact, the experiments show that when the innovation is large (e.g. $0.5m$), $\nabla \mathbf{H}_k$ is fluctuating and changing the sign completely.) The error in $\nabla \mathbf{H}_k$ then propagates directly to $\mathbf{K}$ and then $\hat{\mathbf{d}}_{cr}$, $\hat{\mathbf{P}}_{cr}$. Another source of error comes from $\mathbf{H}(\hat{\mathbf{d}}_r)$ (see (5)), but generally the error in function evaluation is small compared to the error in its Jacobian evaluation. Thus, the RMGF-SA performance is very sensitive to noise error.

The RMGF-IA uses a linear KF as its estimator, which is optimal for linear dynamics/observation models in this case. The largest source of linearization error is from $\mathbf{R}$ in (14) since $\mathbf{R}$ is computed from the Jacobians of h and g. However, the formulation of (14) naturally restrains the "unexpected" effects of $\mathbf{R}$ in case of large linearization error: If $\mathbf{R}$ is erroneously very large, then $\mathbf{K}$ becomes very small, the update (12), (13) has small effect on $\hat{\mathbf{d}}_{cr}$, $\hat{\mathbf{P}}_{cr}$. If $\mathbf{R}$ is erroneously very small, $\mathbf{K}$ can not go unbounded because of the presence of $\mathbf{C}\hat{\mathbf{P}}_r\mathbf{C}^T$. For implementation, one can use a minimum threshold for $\mathbf{R}$. The noise error comes from $\mathbf{b}$ in (12) is small compared to that from $\mathbf{R}$ and $\mathbf{K}$ (similarly to $\mathbf{H}$ in (5)).

Thus, the RMGF-IA is expected to be more robust subject to linearization errors than the RMGF-SA when the observation noise is large. This prediction will be verified with real experiments in the next section.

V. EXPERIMENTS

In the first experiment, Donald Duck, a fully autonomous mobile robot is used. Ten beacons are placed in the environment ($8m \times 6m$). They can be detected as point landmarks by the on-board laser sensors with an accuracy of about $2cm$. The robot makes a complete loop and 1300 observation steps.

Fig.6 shows the estimated robot trajectory and landmark locations using the RMF. The thin curve is the vehicle path using raw odometry data which contain a small systematic error: the raw odometry path does not close. Clearly, the RMF does not use the data coming from the odometry in the estimation, thus the filter is not affected by the odometry error and able to produce a good result. However, a close look reveals 5 inconsistent distances!

To evaluate the consistency of a given relative map, we introduce the following measure. Let's denote d_{ij} as the distance between the ith and j^{th} landmarks. $\hat{d}_{ij}$ is the estimated distance (in $\hat{\mathbf{d}}$). Once the absolute map associated with the estimated $\hat{\mathbf{d}}$ is obtained, it is possible to compute the distance d_{ij}^a from the absolute coordinates of the ith and j^{th} landmarks. If the distance $\hat{d}_{ij}$ is inconsistent then $\hat{d}_{ij} \neq d_{ij}^a$. We introduce the *absolute estimate error (AEE)* as the value of $|d_{ij}^a - \hat{d}_{ij}|$.

In Fig.6, there are 5 inconsistent distances with AEE greater than $1cm$. Fig.7 shows the results using the RMGF: The estimated trajectory is good and there are no inconsistent distances (both RMGF-SA and RMGF-IA produce similar results).

In the second experiment, the algorithms are tested with the Victoria park dataset (a well known dataset in SLAM community). Since the aim of this experiment is to evaluate estimation algorithms, a true data association is used. The first 600 observation steps are selected in this experiment, which include 2 loop closings, 62 (tree) landmarks, 494 (distance) states and covers an area of about $160m \times 100m$. The first two observed landmarks are chosen as the seeding landmarks, and the relative distance between them has
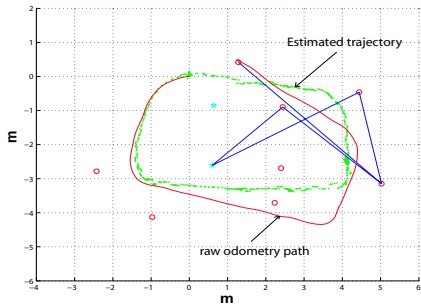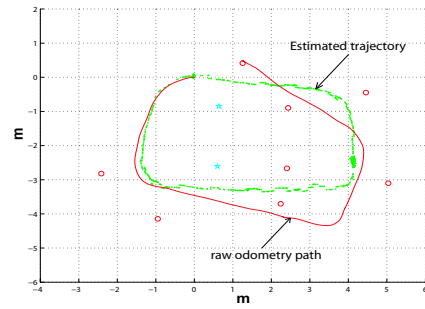

Fig. 7. Estimated vehicle trajectory and landmark locations using the RMGF. There are no inconsistent distances.
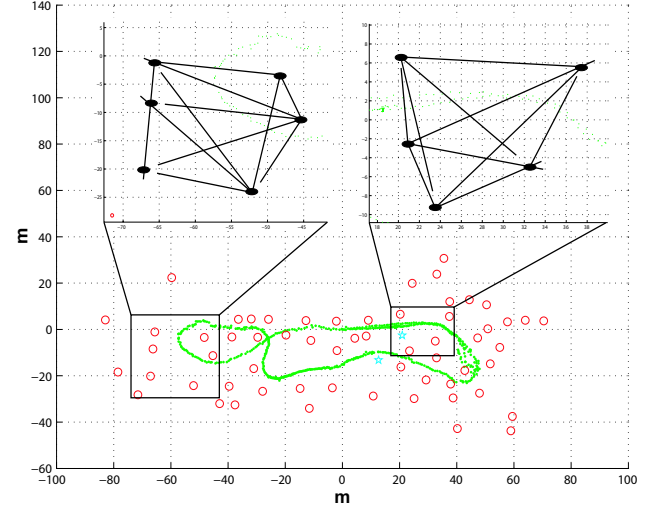

Fig. 8. The RMF on the Victoria park dataset: Bottom - the estimated vehicle trajectory and landmark locations; Top - two zoomed-in examples show inconsistent distances.

an initial variance $\mathbf{P} = [0.5m^2]$. (Note: For this outdoor experiment, we use a value of $(0.5+0.06)m$ as the standard deviation for each newly observed distance, where $0.06m$ is the uncertainty of the hardware and $0.5m$ is the uncertainty for the measured distance from the robot to the center of a tree.)

The result obtained using the RMF is shown in Fig.8. The bottom part shows the estimated vehicle trajectory and landmark locations. The top part displays 2 zoomed-in examples which reveal that some estimated distances are inconsistent. In total, there are 55 inconsistent distances with AEE greater than $10cm$, 4 inconsistent distances with AEE greater than $50cm$ and one inconsistent distance with AEE greater than $1.0m$! It is important to emphasize that in an outdoor application, those distances may not be such great significant for absolute mapping approaches. However, for relative mapping approach, a small local inconsistency can make a big difference in the final map if the map is large, since the error propagates.

We are unable to use the RMGF-SA in this experiment. The problem comes from the linearization. In fact, an IEKF (Iterated EKF) is used in the implementation of RMGF-SA. The values of $\nabla \mathbf{H}_k$ and the innovation between

Fig. 6. Estimated vehicle trajectory and landmark locations using the RMF. There are 5 inconsistent distances (indicated segments).
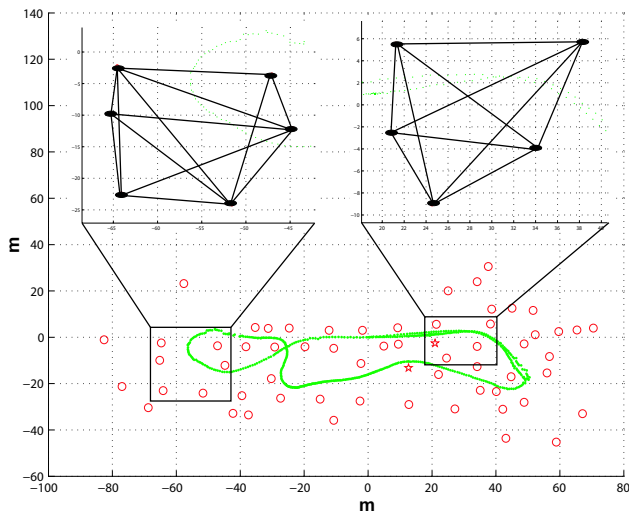
Fig. 9. The RMGF-IA on the Victoria park dataset: Bottom - the estimated vehicle trajectory and landmark locations; Top - two zoomed-in examples. There are no inconsistent distances.

the iterations are fluctuating, not converging. (When the innovation is large (e.g. $0.5m$), $\nabla \mathbf{H}_k$ is fluctuating and changing the sign completely.) After first 100 observation steps, the divergence problem arises which causes negative distance lengths. This can be explained by the large error in the measurement data (the case of Victoria park dataset), making the Jacobian value $\nabla \mathbf{H}_k$ erroneous. Therefore, the linearization does not give a good approximation of the function.

Fig. 9 shows the result obtained using the RMGF-IA. The estimated trajectory is similar but smoother than the one obtained using the RMF. However, there are no inconsistent distances.

Fig. 10 is a comparison between the relative map states estimated by the RMF and those estimated by the RMGF-IA. The figure shows the absolute length differences between the estimated distances by the two algorithms. The median of the differences is about $10cm$. There are 4 peaks where the length differences are greatest. Each of them corresponds to the distances starting from a landmark where the vehicle makes a turn. This is because during a turn, the number of observations of the same set of landmarks is small, thus the correlation between the landmarks in this region is small. As a result, large inconsistency remains.

VI. CONCLUSIONS

This paper has presented two methods to handle the consistency of the Relative Map Filter to solve the SLAM problem. The idea consists in maintaining a geometrically consistent relative map. The first approach, the RMGF-SA interprets the geometric constraint as a perfect observation and adopts EKF as its estimator. The second approach, the RMGF-IA considers the geometric constraint as a virtual observation. It then carries out the estimation by applying a linear KF. Both algorithms scale linearly with the number of landmarks.
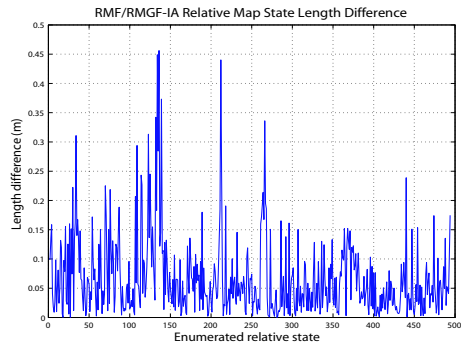

Fig. 10. The difference in length of enumerated relative map states estimated by the RMF and RMGF-IA.

Two experiments on real platform data are performed. The first experiment shows that both methods perform theoretically better than the the original RMF in term of consistency of the estimated map state. However, the RMGF-SA which is based on an EKF, suffers from the linearization errors and does not work when the measurement data are affected by a large noise. That means one has to pay attention when using EKF in real applications. The RMGF-IA, on the other hand, uses a linear Kalman filter and produces very good results in both experiments.

REFERENCES

[1] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan. *Estimation with Applications To Tracking and Navigation*. John Wiley & Sons, Inc., 2001.
[2] M. Csorba, J. K. Uhlmann, and H. F. Durrant-Whyte. A Sub Optimal Algorithm For Automatic Map Building. In *Proceedings of the American Control Conference*, New Mexico, 1997.
[3] M. C. Deans and M. Hebert. Invariant Filtering for Simultaneous Localization and Mapping. In *Proceedings of the International Conference on Robotics and Automation*, volume 2, pages 1042–1047, USA, 2000.
[4] A. Martinelli and R. Siegwart. Improving the SLAM Convergence with a Relative Map Filter. In *Proceedings of the International Conference on Intelligent Autonomous Systems - IAS'2004*, Netherlands, 2004.
[5] A. Martinelli, N. Tomatis, and R. Siegwart. Open Challenges in SLAM: An Optimal Solution Based on Shift and Rotation Invariants. In *Proceedings of the International Conference on Robotics and Automation - ICRA*, 2004.
[6] P. Newman. *On the Structure and Solution of the Simultaneous Localization and Mapping Problem*. PhD thesis, Australian Center for Field Robotics, University of Sydney, 1999.
[7] P. Newman and H. Durrant-Whyte. A New Solution to the Simultaneous Localisation and Map Building (SLAM) Problem - the GPF. In *Proceedings of the SPIE conference*, 2001.
[8] R. Smith, M. Self, and P. Cheeseman. Estimating Uncertain Spatial Relationships in Robotics. In *Autonomous Robot Vehicles*. Springer-Verlag, 1990.