Abstract—In this paper, the Independence Relative Map algorithm is presented. The algorithm aims to achieve the independence of relative map states. We show that using dependent relative quantities from the same observation creates a bias to the state covariance matrix, leading to an inaccurate and inconsistent algorithm. The algorithm is presented. The algorithm aims to achieve the independence of relative map states. Having independent map states improves the map consistency. Two case studies are presented in which we apply the proposed algorithm together with two popular relative map methods. Experimental results on simulated data show that the integrated algorithms outperform the original methods in term of map consistency and algorithm speed.

Keywords: Consistency, Relative Map, SLAM.

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.

In traditional SLAM (see [1]), it is common to use the absolute map approach. However, one of the drawbacks is that this approach is based on a perfect statistical knowledge of the errors of the robot sensors (e.g. odometry) and also on a hypothesis of linear observation which are generally not the case for real world applications. Another possible approach is to use the concept of relative map that has better convergence properties than the absolute map approach [2]. 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 is given in [3]. Lately, [4] introduces a relative filter which is based on quantities invariant to the robot pose, i.e. to shift and rotation. The same idea is adopted in [5]. Both algorithms estimate the relative distances between features pairwise.

Another relative map algorithm has been proposed few years ago in [6], called Geometric Projection Filter - GPF. It consists of two parts: the first part is essentially based on the relative filter introduced in [4]. The second part is a consistency enforcement which provides a means to produce a geometrically consistent map from the estimate of the first part by applying a set of linear constraints among the map elements in a Kalman filter. However, the elements used in this algorithm are invariant to shift only, not to rotation.

Recently, [2] extends the relative filter introduced in [4] so that the new algorithm also takes into account the correlation between the map states, e.g. distances between point features. The estimation is then carried out by applying a Kalman filter. The same author group later introduce an algorithm in [7], called Relative Map Geometric Filter - RMGF, in order to maintain the consistency of the relative map estimate by enforcing the geometric constraints between the map states.

This paper aims to further improve the consistency property of a relative map. The key idea is that some information is reused when dependent relative quantities are used from the same observation. A proper way of selecting independent relative quantities is then proposed.

The paper is organized as follows. The next section gives an example illustrating the problem of information reuse by using dependent relative quantities. Section III gives a brief review of the RMGF. For a full description of the GPF please refer to [6]. Next, the Independence Relative Map procedure is introduced. Two adapted versions for the GPF and RMGF are proposed in the followed case studies. The experimental results are presented in section VI with discussions. Finally, some conclusions are drawn.

II. THE PROBLEM OF RELATIVE MAP CONSISTENCY

The problem of map consistency has been addressed in the SLAM literature [8], [9], [6]. It can be divided into two main types of the consistency of the map state. The first type of consistency (or inconsistency problem) is due to the approximations from the linearization of the dynamics and observation equations that can lead to divergence of the EKF [10], [8], [9]. As a result, there is no guarantee that the computed covariances will match the actual estimation error. In this paper, we do not consider this type of consistency.

The second type of consistency concerns only a relative map [6], [7]. The following definition is adopted from [6]:

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 one known landmark (for the case of GPF) or from two known landmarks (for the case of RMGF) and produces a unique solution for each landmark location, then the relative map is consistent. It has been shown in [6], [7] that by imposing the constraints among the relative map states, one can improve the consistency of the resulting map.
We will show in the following example that by using dependent relative quantities from the same observation, one may create a bias to the relative map state, resulting in an inaccurate and inconsistent map state.

Example: Assume we have one observation consisting of the measurements of absolute locations of three 1D point features \(x_1, x_2, x_3\). The measured values are \(x_1 = (a, \sigma^2)\), \(x_2 = (b, \sigma^2)\), \(x_3 = (c, \sigma^2)\). Using the relative map approach, we compute the relative quantities (i.e. distances):

\[
d_{12} = x_2 - x_1 \quad ; \quad d_{23} = x_3 - x_2 \quad ; \quad d_{13} = x_3 - x_1
\]

We call the measured values of \(x_1, x_2, x_3\) the primary measurements and the deduced values of \(d_{12}, d_{23}, d_{13}\) secondary measurements (or relative quantities). We have the relative map state as follows:

\[
d = \begin{bmatrix} d_{12} \\ d_{23} \\ d_{13} \end{bmatrix} = \begin{bmatrix} b-a \\ c-b \\ c-a \end{bmatrix}
\]

\[
P = \begin{bmatrix} 2\sigma^2 & 0 & 0 \\ 0 & 2\sigma^2 & 0 \\ 0 & 0 & 2\sigma^2 \end{bmatrix}
\]

Now taking the constraint \(d_{13} - d_{12} - d_{23} = 0\) or in vector form \(z = \mathbf{C}d\) as a perfect observation where \(\mathbf{C} = [-1 \ -1 \ 1]\) and applying the Kalman filter, we have:

\[
d' = \begin{bmatrix} b-a \\ c-b \\ c-a \end{bmatrix}
\]

\[
P' = \begin{bmatrix} \frac{4}{3}\sigma^2 & -\frac{2}{3}\sigma^2 & \frac{2}{3}\sigma^2 \\ -\frac{2}{3}\sigma^2 & \frac{4}{3}\sigma^2 & -\frac{2}{3}\sigma^2 \\ \frac{2}{3}\sigma^2 & -\frac{2}{3}\sigma^2 & \frac{4}{3}\sigma^2 \end{bmatrix}
\]

Now from the new covariance matrix \(P'\), the new variance of \(d_{23}\) is \(\frac{4}{3}\sigma^2\). However, from the beginning after receiving the observation measurements of \(x_1, x_2\) and \(x_3\), it is obvious that the value of \(d_{23} = x_3 - x_2\) is \((c-b, 2\sigma^2)\). Thus, the computed variance of \(d_{23} = \frac{4}{3}\sigma^2\) after applying the Kalman filter is smaller than the actual estimation error, meaning the relative map is inconsistent. The reason is that some information is reused when all the three relative quantities are considered as the new relative measurements.

Notice that if \(d_{13}\) is obtained from another observation (different primary measurements) then \(d_{13}\) can be added to the map state without creating a bias. In this case, the correlation (or dependency) between \(d_{13}\) and \(d_{12}, d_{23}\) are enforced by imposing the constraint as in GPF or RMGF.

On the other hand, if we consider the correlation between the relative quantities at the (secondary) measurement level, we will have:

\[
d = \begin{bmatrix} d_{12} \\ d_{23} \\ d_{13} \end{bmatrix} = \begin{bmatrix} b-a \\ c-b \\ c-a \end{bmatrix}
\]

\[
P = \begin{bmatrix} 2\sigma^2 & -\sigma^2 & \sigma^2 \\ -\sigma^2 & 2\sigma^2 & -\sigma^2 \\ \sigma^2 & -\sigma^2 & 2\sigma^2 \end{bmatrix}
\]

We remark that the determinant of \(P\) is 0. This reflects that the constraint among the elements has been inclusively imposed (the elements in the map are dependent). However, in a general case the relations among the primary and secondary measurements are not linear, meaning that the elements of \(P\) are computed by using the first order Taylor approximation. Thus, the structure of \(P\) is verified the constraint only approximately. A proper way to minimize this problem is to consider only independent elements in the same observation.

There are two questions that one may ask:

1) Is it possible to obtain a relative map estimate that contains only independent quantities without using any geometric constraint enforcement?

2) If it is not possible, how do we maintain the independence of the map state and at the same time enforce the map consistency?

If it is the case for the first question, then we do not need any consistency filter, thus reducing the computational complexity and still being able to obtain a consistent relative map. In this paper, we will try to give answers to these two questions.

III. THE RELATIVE MAP GEOMETRIC FILTER REVIEW

In the RMGF [7], 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 \(d\) as the state and \(P\) as its covariance matrix. In Fig.1a, the vector \(d\) contains the indicated distances between the 6 landmarks. Clearly, not all of the distances between the 6 landmarks are stored in \(d\) 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). Notice that all the pairwise distances are considered in the current observation. These distances may have been observed (i.e. already in \(d\)) or may not. Let’s introduce the following notations:

\[
d_{\text{old}} = [u, w_{\text{old}}]^T 
\]

\[
d_{\text{obs}} = [w_{\text{obs}}, v]^T
\]

where \(d_{\text{old}}\) is the state estimated at a given time step just before a new observation is made; \(d_{\text{obs}}\) is the observation at the same time step, containing a set of distances between the landmarks observed by the robot. \(u\) contains the distances which are not re-observed (i.e. which do not appear in the vector \(d_{\text{obs}}\)). \(w_{\text{old}}\) contains the distances re-observed (denoted by \(w_{\text{obs}}\) in the vector \(d_{\text{obs}}\)). Finally, \(v\) contains the distances observed for the first time at the considered time step. The associated covariance matrices are:

\[
\begin{bmatrix} P_{\text{old}} \\ P_{\text{obs}} \end{bmatrix} = \begin{bmatrix} P_{uu} & P_{uw} \\ P_{wu} & P_{ww} \end{bmatrix}
\]

\[
\begin{bmatrix} R_{uw} & R_{uv} \\ R_{wu} & R_{vv} \end{bmatrix}
\]

Fig.2 shows the structure of RMGF. In the first stage, only the relative map filter - RMF [2] is applied in fusing the old map elements with the newly observed ones. We adopt the following notations of the quantities estimated by the RMF:
We can obtain the intermediate estimates of the state and its covariance matrix by applying a Kalman filter (refer to [2] for the decomposed equations). Notice that the observation is linear in the state (is the identity) and therefore the Kalman filter is optimal. In the second stage, the inconsistency of the relative map estimate from the first state is analyzed and eliminated by a consistency constraint enforcement (Fig.2).

Fig.3a depicts an example of such inconsistency. The absolute location of landmark x\(_3\) can be recovered from the previously computed locations of landmarks x\(_1\), x\(_2\), x\(_3\) and the estimated distances d\(_{12}\), d\(_{23}\), d\(_3\). However, due to the imperfection in the estimation, the relative transformation results in two inconsistent solutions: one obtained from the set \{x\(_1\), x\(_2\), d\(_{12}\), d\(_2\)\} and one obtained from \{x\(_2\), x\(_3\), d\(_{23}\), d\(_3\)\} (the third solution obtained from \{x\(_3\), x\(_1\), d\(_{31}\), d\(_1\)\} is independent and can be deduced from the other two solutions). The inconsistency of the relative map estimate emerges when recovering the absolute map.

The key idea is to use each landmark location as a fusing point to enforce the consistency of the estimated relative map. Again in Fig.3a, if we are able to fuse or unite the two inconsistent solutions, we will obtain a unique consistent solution for landmark 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 without any ambiguities.

In [7], two methods are proposed. Here we describe only the first method to illustrate the technique. In the implementation we use the second method since it has been shown to be more reliable to linearization errors. In Fig.3b, we can make the following reasoning: “fusing” x\(_{12}\), x\(_{23}\) is equivalent to equalizing x\(_{12} = x_{23}\) or the constraint x\(_{12} - x_{23}\) = 0 must be satisfied. The fact that x\(_{12}\) is a function of x\(_1\), x\(_2\), d\(_{12}\); and x\(_1\), x\(_2\) are considered fixed (given or already computed in previous steps), x\(_{12}\) is a function of d\(_{12}\), d\(_2\).

Similarly for x\(_{23}\). In vector form, we can write the constraint as \(H(d_{rf}) = 0\). If we interpret the constraint as a perfect observation \(z = H(d_{rf})\), applying the Kalman filter we have:

\[
\begin{align*}
\dot{d} &= d_{rf} + K (0 - H(d_{rf})) \\
P &= P_{rf} - K \nabla H P_{rf} \\
K &= P_{rf} \nabla H^T [\nabla H P_{rf} \nabla H^T]^{-1}
\end{align*}
\]

where \(d_{rf}\) and \(P_{rf}\) are the map state and covariance matrix estimated from the first stage, \(d\) and \(P\) are the estimate and corresponding covariance matrix of the consistent relative map. The absolute map can be recovered uniquely from the relative map estimate by a relative transformation, given the absolute locations of two seeding landmarks.

**IV. THE INDEPENDENCE RELATIVE MAP**

Our first question is “Is it possible to achieve a relative map estimate that contains only independent quantities without using any geometric constraint enforcement?”. If it is the case then we do not need any consistency filter, thus reducing the computational complexity and still being able to obtain a consistent relative map.

An counter example is shown in Fig.4. The relative quantities are distances between point features. The current relative map in Fig.4a contains only independent quantities. Even though the new observation (Fig.4b) contains only independent quantities, it has a different element than those already in the map. As a result, the new map obtained in Fig.4c contains dependent quantities. It is in general the case that when the robot changing its view of observation in subsequent steps or making a loop closing, the newly observed elements are formed differently to the past. Thus, the dependency between the estimate quantities is not avoidable in general.

By the time of writing this paper, there is an independent work [11] in which the authors propose an algorithm using the relative map approach. In particular, distances and angles between point features are used as relative quantities. It states that the state vector always contains a minimal, thus independent, number of map elements \(2n - 3\) for \(n\) point features in 2D. It is easy to show that the claim does not hold. A counter example is shown in Fig.5. The newly obtained relative map contains dependent quantities in Fig.5c. (The examples discussed in [11] use a presumption that it is always possible to observe seeding features 1 and 2, which helps, but it is not practical in general.) To answer the second question “how do we maintain the independence of the map state and at the same time enforce the map consistency?”, we introduce the Independent Relative Map procedure that can be applied to any relative map strategy. It consists of three parts:

- Extracting independent relative quantities (relative measurements) from one observation.
- Consistency enforcement for dependent quantities (those are generated from different observations).
• Dependent elements removal (after consistency enforcement).

For the first part, we recall the example in shown Fig.1b. Instead of taking all the 6 distances between the 4 landmarks as new relative quantities, we can equivalently consider 5 distances. By considering only independent relative quantities in an observation, which is minimal for the observation without any information loss, we can prevent creating a bias to the map state. Furthermore, doing that will decrease the number of map elements and also minimize the possibility of creating new constraints, thus reducing the work needed for the consistency enforcement filter. In the Case Study, we will show in more detail for the two selected cases. Fig.6 shows the structure of the Independence Relative Map - IRM algorithm. It consists of three main components. The first one is simply a relative filter algorithm (e.g. GPF, RMGF) which generates an unconstrained relative map. The choice of relative quantities is algorithm dependent. The only difference is that each new “relative measurements” should contain only independent quantities. The second component is a consistency enforcement where the constraints are applied. The method is dependent on the type of relative quantities (see [6], [7]). The output of the consistency filter is a consistent relative map, from which one can generate a consistent absolute map. The dependent elements are removed from the state vector by the third component.

Having considered only independent relative quantities from one observation leads to reducing the size of the state vector and minimizing the number of constraints to be applied. In other words, the computation cost is reduced. There are several strategies to select independent elements in an observation. For example, it is advantageous to select already existing elements or ones formed by features in proximity so that they will likely be observed again in following steps.

For the linear case (as in GPF), the linear constraints should be applied once only [12]. After that the dependent elements can be removed without any information loss. The removal can be performed in between the observation steps or after the last step. For the nonlinear case (as in RMGF), due to the approximation of the EKF, the constraint enforcement can be performed iteratively (periodically or for several steps) until a convergence condition is met. After that, the constraints and the dependent elements can be removed.

V. CASE STUDY

A. Case study 1: The Independence RMGF

Applying the IRM procedure to the RMGF algorithm is straightforward. Fig.7 shows an example of the working of the Independent RMGF. In Fig.7b, instead of considering all the distances as in the original RMGF, we consider only $2 \times 4 - 3 = 5$ independent distances. (There are $2 \times n - 3$ independent relative distances in an observation containing $n$ 2D-point features.) Notice that since the distance $d_{34}$ has not been observed or considered so far, fusing the new observation to the existing map generates an dependent distance (Fig.7c).

Constraint construction is performed similarly as it is done in RMGF: Start from a graph $g$ (of point landmarks and distances) consisting of two given seeding landmarks. Graph $g$ is kept always connected. Each step, one landmark is added to $g$ at a time. The number of constraints related to this landmark equals the number of distances linking the landmark to $g$ minus 2. Next, the constraints are constructed and applied by the consistency filter as described in [7]. Dependent elements are removed periodically or after the last step.

B. Case study 2: The Independence GPF

In this case study, we apply the IRM procedure to the GPF. For a full description of GPF, please refer to [6].

Fig.8 shows a working example of the Independent GPF. In Fig.8b, instead of considering all possible pairings in a new observation, only independent vectors (relative quantities) are stored (3 vectors from 4 landmarks). The relative map after fusing the observation is depicted in Fig.8c which contains one dependent vector. This is because the landmarks 3 and 4 have been already observed, but no direct vector between them has been observed or added to the state vector. At this point, one can apply the consistency filter to enforce the constraint.

In an observation (or a connected map) containing $n$ 2D-point landmarks, the number of independent vectors is $n - 1$ and they do not contain a loop. The strategy to select the elements is similar to the previous case: choose the already existing vectors first, otherwise select the vectors between proximate landmarks.

\[ \begin{align*}
&\text{Independent Relative Quantities} \\
&\text{Dependent Element Removal} \\
&\text{Unconstrained Relative map} \\
&\text{Consistency Constraint Enforcement} \\
&\text{RF} \\
&\text{Consistent Relative map} \\
&\text{Consistent Absolute map}
\end{align*} \]
The number of constraints equals the number of dependent vectors. To find and construct the constraints, one can follow a procedure similarly to the one when recovering the absolute map from a relative map: Start from a graph $g$ (of landmarks and vectors) consisting of a given seeding landmark. Graph $g$ is kept connected and containing no loop. Each step, one landmark is added to $g$ at a time. The number of constraints related to this landmark equals the number of vectors $n_v$ linking the landmark to $g$ minus 1 (one vector is sufficient to uniquely determine the landmark). Add one vector to $g$. For each of the remaining $n_v-1$ vectors, a constraint is constructed by finding the closed path (there exists only 1) between the landmark and $g$ containing the vector. Sum of the vectors forming the loop equals zero. After applying the constraint, the vector is removed. The process is repeated for all $n_v-1$ vectors of the landmark. Similarly for other landmarks. The resulting graph $g$ is connected and contains no loop, therefore is not subjected to any geometric constraint. The complexity of this procedure is $N_c \times \log N$ ($N_c$ is the number of constraints applied, $N$ is the number of landmarks).

In the GPF, the relative elements are considered uncorrelated to each other. However, one can improve the optimality of the filter by considering the correlation between each pair of vectors sharing a common landmark.

VI. EXPERIMENTAL RESULTS

To evaluate the performance of the filters using the IRM algorithm, we have implemented two classes of algorithms: the first class uses the GPF relative map algorithm (i.e. relative quantities are vectors), the second one uses the RMGF algorithm (i.e. relative quantities are distances). Point landmarks are used as map features. For each class, we also consider 2 cases: the observation covariance matrix is diagonal or fully correlated.

In order to test the estimation performance of the algorithms, we use a simulated map and sensors measurements in the experiment. By doing this we can avoid the problem of data association, i.e. the ground truth is known. The map is generated to have a similar structure as the map of Victoria Park dataset. The map contains 76 point landmarks. There are 1000 observation steps. The green thick line is the true robot trajectory, the red thin line is the odometry trajectory.

We use 3 following measures to evaluate the algorithms:

- Median of the length differences of the computed relative quantities compared to the ground truth.
- Median of the discrepancies of the landmark locations in the recovered absolute map compared to the ground truth.
- Running time.

Note that the minimal/maximal values of the discrepancy are likely corresponding to the results of the first/last observation. Therefore, the median value is used to reflects the average discrepancy after 1000 observation steps.

Table I and II show the results of the two algorithm classes. The prefix “I” denotes the algorithm using the IRM procedure. The algorithms with (c) have the fully correlated observation covariance matrix. The algorithms IGPF and IRMGF generate the correct numbers of independent relative elements after the dependency removal (150 and 149 respectively).

The first observation is that the discrepancies of relative quantities are much better for the algorithms using the IRM. In other words, the relative maps generated using the IRM are more consistent.

In term of absolute landmark locations, the errors are much smaller for the IRMGF compared to those produced by the RMGF. There are two main reasons for this. First, the con-
number of independent elements are computed correctly, being

dents. Moreover, the increase in running time is very small.

than their correspondents. This is because their state vector

consistency lters at the end of the algorithms, therefore

has much smaller size. In this implementation, we apply the

numbers of estimated states shown in the tables reect the size

for the case of

being imposed, the more approximation errors being generated

the rst algorithm class where the lters are linear. Second,

for the case of RIMGF(c) where the observation covariance

matrix is fully correlated, meaning the constraint is inclusively

imposed, the matrix elements are approximated by the rst

order term of the Taylor expansion. Using independent relative

quantities will remove the rows/columns in the observation

covariance matrix corresponding to the constraints. Therefore,

effects of the approximation is minimized.

Observe that the algorithms implemented using fully cor-

related covariance matrix perform better than their corre-

spondents. Moreover, the increase in running time is very small.

In term of running time, the algorithms using the IRM

procedure always perform in one order of magnitude faster

than their correspondents. This is because their state vector

has much smaller size. In this implementation, we apply the

consistency lters at the end of the algorithms, therefore

the dependent quantities are only removed after that. The

number of independent elements are computed correctly, being

150 = 76 × 2 − 2 and 149 = 76 × 2 − 3 for each case. (The

numbers of estimated states shown in the tables reect the size

of the state vector during the estimation before the dependency

removal.)

We can not compare the results obtained from the two
classes since for the algorithms in the rst class we have
some extra information about the global angles of the relative
vectors. In general, we expect that the algorithms in the rst
class would produce better results because the Kalman filter
is optimal for linear systems. However, it is quite surprising
that the absolute map obtained by using the RIMGF is much
better than the ones obtained by the rst class algorithms
where they have the optimal estimator for the linear relative
filters and linear consistency lters. In other words, in this
experiment setting, the RIMGF is able to perform better than
the linear algorithm class which use extra information on
absolute direction within an error of \( \sigma^2_{\phi} = 0.5 \text{deg}^2 \) (this
precision of a compass is very difcult to achieve in practice).

VII. CONCLUSION

In this paper, we have presented the Independence Relative
Map algorithm that can be integrated to a relative map
algorithm. The advantage of the algorithm is two folds. First,
by using only the independent relative quantities in the same
observation, the bias to the covariance matrix is minimized or
removed. This improves the consistency of the relative map
estimate and in turns generates a more accurate absolute map.
Second, by considering only independent relative quantities
from the same observation and combining with dependent
elements removal, a minimal number of elements are included
in the map vector and therefore the computational cost is
signifcantly reduced.

We have applied the IRM procedure to the two relative
map algorithms GPF and RIMGF. We have implemented and
empirically verifed the performance of the algorithms on the
simulated data. The results show signifcant improvement in
terms of relative map consistency, absolute map accuracy and
algorithm speed.

ACKNOWLEDGMENTS

This work has been supported by the Swiss National Science
Foundation No 200021-101886 and the EU project Cogniron
FP6-IST-002020.

REFERENCES

[1] S. Thrun, Y. Liu, D. Koller, A. Ng, Z. Ghahramani, and H. Durrant-
Whyte, “Simultaneous Localization and Mapping with Sparse Extended
Information Filters,” International Journal of Robotics Research, vol. 23,
SLAM: An Optimal Solution Based on Shift and Rotation Invariants,” in
ization and Mapping Problem,” Ph.D. dissertation, Australian Center for
Field Robotics, University of Sydney, 1999.
[8] S. Julier and J. Uhlmann, “A counter example to the theory of simultane-
ous localization and map building,” in Proceedings of the International
of EKF-Based SLAM,” in Proceedings of the 5th IFAC Symposium on
[10] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, Estimation with Applica-
Mapping in SLAM Using Compact Relative Maps,” in Proceedings of
IROS’2005.
Smoothly Constrained Kalman Filter,” IEEE Transactions on Pattern

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>N. States Estimated</th>
<th>Rel. Dist. Difference (cm)</th>
<th>Abs. Pos. Discrepancy (cm)</th>
<th>Running Time (second)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPF</td>
<td>2000</td>
<td>7.8</td>
<td>21.2</td>
<td>1315</td>
</tr>
<tr>
<td>GPF (c)</td>
<td>2000</td>
<td>7.4</td>
<td>20.7</td>
<td>1372</td>
</tr>
<tr>
<td>IRGF</td>
<td>260</td>
<td>3.4</td>
<td>23.5</td>
<td>118</td>
</tr>
<tr>
<td>IRGF (c)</td>
<td>260</td>
<td>3.0</td>
<td>20.5</td>
<td>117</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>N. States Estimated</th>
<th>Rel. Dist. Difference (cm)</th>
<th>Abs. Pos. Discrepancy (cm)</th>
<th>Running Time (second)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIMGF</td>
<td>1000</td>
<td>9.9</td>
<td>75.2</td>
<td>1214</td>
</tr>
<tr>
<td>RIMGF (c)</td>
<td>1000</td>
<td>4.8</td>
<td>22.8</td>
<td>1378</td>
</tr>
<tr>
<td>IRMGF</td>
<td>265</td>
<td>6.3</td>
<td>31.5</td>
<td>150</td>
</tr>
<tr>
<td>IRMGF (c)</td>
<td>265</td>
<td>4.4</td>
<td>11.9</td>
<td>147</td>
</tr>
</tbody>
</table>