#### Robotics and Autonomous Systems 60 (2012) 83-94



## **Robotics and Autonomous Systems**

journal homepage: www.elsevier.com/locate/robot



# Inferring laser-scan matching uncertainty with conditional random fields\*

Zuolei Sun<sup>a,\*</sup>, Joop van de Ven<sup>b</sup>, Fabio Ramos<sup>b</sup>, Xuchu Mao<sup>c</sup>, Hugh Durrant-Whyte<sup>b</sup>

<sup>a</sup> College of Information Engineering, Shanghai Maritime University, Shanghai, 200135, China

<sup>b</sup> ARC Centre of Excellence for Autonomous Systems, Australian Centre for Field Robotics, University of Sydney, 2006, NSW, Australia

<sup>c</sup> School of Electronic, Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China

#### ARTICLE INFO

Article history: Received 20 May 2010 Received in revised form 6 September 2011 Accepted 8 September 2011 Available online 8 October 2011

Keywords: Laser scan matching Conditional random fields (CRF) Covariance matrix Simultaneous localisation and mapping (SLAM)

#### ABSTRACT

This paper proposes a novel algorithm for computing robot motion estimates from ranging sensors. The algorithm utilises the recently proposed CRF-Matching procedure which matches laser scans based on shape descriptors. The motion estimates are computed in a sound probabilistic framework by performing inference on a probabilistic graphical model. The Sampling-Product inference algorithm is proposed for obtaining probable association hypothesis from the probabilistic model. The hypothesis are used to generate estimates on the uncertainty of translational and rotational movements of the mobile robot. Experiments demonstrate the benefits of the approach on simulated data sets and on laser scans from an urban environment. The approach is also combined with the well-established delayed-state information filter for a large-scale outdoor simultaneous localisation and mapping task.

© 2011 Elsevier B.V. All rights reserved.

#### 1. Introduction

Reliable navigation in mobile robotics requires the computation of robust motion estimates. Solutions based on inertial measurement units or GPS can provide an estimate and corresponding uncertainties directly. For ranging sensors however this task is significantly more complex. The difficulty lies in obtaining robust point correspondences between consecutive scans from which the motion estimates are computed. Solutions based on the Iterative Closest Point (ICP) [1,2] can, in general, provide reasonable motion estimates. However, for reliable navigation with a scanmatching procedure [1], deterministic estimates of the motion are not enough; it is also necessary to quantify the uncertainty on these estimates.

The motion estimates' uncertainty, computed from range sensors, arrives from two main sources: (1) uncertainty in the point associations; (2) uncertainty in the range and bearing measurements from the sensor. For the case of laser range finders, range and bearing estimates are very accurate and their uncertainty is almost insignificant compared to the uncertainty from the point association. The computation of uncertainty from wrong point associations is much more challenging as it involves the evaluation of an enormous set of possible associations. For example, in a conventional data association problem, with a pair

\* Corresponding author.

of laser scans with 361 laser points each, the association space is  $\mathcal{O}(361^{361})$ . The probability of a particular scan association is then computed by evaluating the likelihood of the association divided by the sum of the likelihoods of all possible associations. This computation quickly becomes unfeasible, even for laser scans with a relatively small number of points.

Uncertainty quantification of scan matching is a significant problem for mobile robotics using laser range finders. The problem has received much attention since Lu [1] first addressed it. Most solutions focus on ICP like algorithms and can be categorised according to the uncertainty sources they address; measurement noise, lack of pairwise constrains, local minima, and improper laser point associations. Related work for each of the four categories will be discussed next.

Intuitively, measurement noise from imperfect sensors adds uncertainty. The general approach to dealing with such isotropic or non-isotropic sensor noise is by means of the *likelihood* function. In [3, Section VI.A], the uncertainty is modelled as a *likelihood* constructed from a multi-Gaussian distribution accounting for both inliers and outliers. The individual Gaussians are parameterised by the covariance of range measurements with proper linearisation. Analogously, [4] attempts to reveal the relationship between measurement noise and the uncertainty estimate from which a weighted matching algorithm is proposed. Wang [5, Section 3.3] again employs the *likelihood* function but for a grid-based approach. Other approaches, such as [6,7], [1, Section 6.2], take advantage of *Laplace*'s method to approximate the sensor noise distribution by a Gaussian.

Non-Gaussian likelihood functions are approximated only by their first two moments; the mean and the variance. This is

<sup>&</sup>lt;sup>†</sup> This work is supported by the Science and Technology Program of Shanghai Maritime University (No. 20110025), the National Natural Science Foundation of China (No. 61105097).

E-mail address: sunzuolei@gmail.com (Z. Sun).

<sup>0921-8890/\$ -</sup> see front matter © 2011 Elsevier B.V. All rights reserved. doi:10.1016/j.robot.2011.09.001

feasible, and reasonable, only when sufficient associated laser point pairs are obtained from consecutive poses. Then the true uncertainty is highly peaked and the approximation from non-Gaussian to Gaussian is accurate [3]. Conversely, it is possible there are very few correspondences due to little overlap between scans or a lack of orthogonal constraints. In such cases the *likelihood* is illrepresented by a Gaussian [3]. Furthermore, in case when only 1 or 2 associations are found, minimising the error function for ICP (or its variations) is a high risk venture [7]. Typical under-constrained situations are the corridor and circular environments.

ICP optimises a non-convex error function and usually finds a local minima. As shown in [3, Section VI.C], one way to avoid local minima is to adequately search the relevant space of the likelihood function by drawing a set of samples from the prior distribution of the relative transformation. Each sample is used to initialise a separate run of ICP, each converging to a (perhaps) different local minima. The covariance estimate is then computed from the combination of these solutions. One such application is explored in [5].

As mentioned above, it is generally accepted that for high accuracy sensors, such as scanning lasers, improper associations are much more critical compared to the above three uncertainty sources. Unfortunately, most solutions are based on the assumption that a perfect association is available, and association uncertainty modelling is seldom treated thoroughly. The Laplace method naturally does not account for association uncertainty [3]. The weighted method relies on the correct estimation of parameters [4]. While the sampling-based *offline* approach proposed in [6] is somewhat *ad hoc*.

The recently proposed CRF-Matching algorithm [8] tackles the problem of laser scan association by formulating it as a probabilistic inference procedure in a graphical model. This allows the incorporation of general features that take into account shape descriptors to match the scans. With such a formulation, the laser point association problem can be addressed in a integrated fashion, reasoning over the space of all associations. This paper extends the former by providing a procedure to compute relative motion estimates from probable scan associations. In particular, the main contribution presented here is the development of a novel inference algorithm for probabilistic networks, Sampling-Product inference, that efficiently seeks probable configurations in the space of laser point associations.

The paper is organised as follows. The following section defines the uncertainty quantification problem that we aim to solve. Section 3 discusses the implementation details of laser scan matching with CRF. Section 4 presents the Sampling-Product inference algorithm. Section 5 examines the performance of the algorithm in regards to simulated and real data. Discussions and conclusions are given in Sections 6 and 7 respectively.

### 2. Problem definition

This section defines the problem addressed in the paper. Generally a laser scan is acquired from a rotating laser beam emanating from a sensor mounted on a mobile robot. It measures the range from the sensor to objects in the environment and produces a 2D slice of the environment. The range sequences can be mapped to Cartesian coordinates by defining a coordinate frame in which the pose of the laser scanner is employed as the origin. As shown in Fig. 1, suppose that the robot measures the reference laser scan  $\mathbf{L}_{ref}$  at pose  $P_{ref}$  and acquires a new scan  $\mathbf{L}_{new}$  after moving to pose  $P_{new}$ . Here we assume that the sensor-centric coordinate frame coincides with the robot-fixed frame. The consecutive scans can be formulated as sequences of laser points specified in a Cartesian system:



**Fig. 1.** The laser scan acquisition scenario. The triangles denote robot's consecutive poses  $P_{ref}$  and  $P_{new}$  at which the two laser scans  $L_{ref}$  and  $L_{new}$  are acquired.



**Fig. 2.** Graphical illustration of motion uncertainty introduced by improper laser point associations. Here both the reference and new laser scans are assumed to consist of four points. Only ten possible association configurations are visualised in the example, while the full association space is  $\mathcal{O}(5^4)$ .

$$\mathbf{L}_{\text{ref}} \equiv \left\{ \left( L_{\text{ref}_{x}}^{i}, L_{\text{ref}_{y}}^{i} \right) \right\}_{i=1}^{N}$$
(1)

$$\mathbf{L}_{\text{new}} \equiv \left\{ \left( L_{\text{new}_{X}}^{i}, L_{\text{new}_{y}}^{i} \right) \right\}_{i=1}^{M},$$
(2)

where *N* and *M* are the number of laser points in the reference and new scan respectively. Assuming the pose difference is not very significant, the association between the overlapping parts of  $\mathbf{L}_{ref}$  and  $\mathbf{L}_{new}$  results in an estimate of the robot's rotation and translation, i.e.,  $\mathbf{RT} = [\mathbf{R}, \mathbf{T}_x, \mathbf{T}_y]$ . The estimate has the following spatial relationship,

$$P_{\text{new}} = P_{\text{ref}} \oplus \mathbf{RT},\tag{3}$$

where  $\oplus$  denotes the compounding operation [9].

The laser point association problem can be defined as follows; laser points in the reference scan are to be associated to their most likely counterpart in the new scan. Ideally a reference point and its associated new point correspond to the same physical point in the real world, such as  $L_{ref}^3$  and  $L_{new}^2$  in Fig. 1. If a point in the reference scan is not able to be associated, then it will be considered an outlier, like  $L_{ref}^1$  in Fig. 1. Utilising the most likely laser point associations to estimate the relative rotation and translation between consecutive poses is referred to as *scan matching* or *scan alignment* [1,2]. The classic scan matching algorithm, Iterative Closest Point (ICP), is widely used in the field of robot navigation.

As addressed in Section 1, the primary cause for the uncertainty of the rotation and translation solution are improper laser point associations. Fig. 2 explains the transformation from possible association configurations to the uncertainty of motion estimates. The relative motion solutions **RT**, calculated from the configurations, are approximated by a Gaussian distribution. The uncertainty, parameterised by the covariance matrix, can then be inferred. The challenge of uncertainty quantification lies in the huge association space which requires  $\mathcal{O}((M + 1)^N)$  evaluations.

#### 3. Laser scan matching with CRF

#### 3.1. Conditional random fields

A Conditional Random Field (CRF) [10] is a *probabilistic discriminative framework* which is usually represented using an undirected graphical model. Graphical models represent probability distributions; the vertices of the graph index the distribution's random variables, while the edges of the graph capture relationships between variables. The formal definition of a CRF is given in [10,11].

**Definition 1.** Let  $G = (\mathcal{V}, \mathcal{E})$  be a graph such that **x** is indexed by the vertices  $\mathcal{V}$  of *G*. Then (**x**, **z**) is said to be a conditional random field, if, when conditioned on **z**, the random variables  $x_i$  obey the Markov property with respect to the graph:  $P(x_i | \mathbf{z}, \mathbf{x}_{\mathcal{V}-i}) =$  $P(x_i | \mathbf{z}, \mathbf{x}_{\mathcal{N}_i})$ , where  $\mathcal{E}$  refers to the edges in  $G, \mathbf{x}_{\mathcal{V}-i}$  is the set of all the nodes in the graph except the node  $x_i$ ,  $\mathcal{N}_i$  is the set of neighbours of the node  $x_i$  in *G*, and  $\mathbf{x}_{\Omega}$  represents the set of labels at the nodes in set  $\Omega$ .

According to the definition, the hidden variables  $\mathbf{x} = \langle x_1, x_2, ..., x_N \rangle$  are globally conditioned on the observations  $\mathbf{z}$ . This allows us to integrate the overlapping observations ( $\mathbf{L}_{ref}$  and  $\mathbf{L}_{new}$ ) into the model. The definition also specifies requirements on the types of distributions that can be used with a CRF. The distribution must obey the Markov property with respect to the graph. Meaning each hidden variable  $x_i$  must be independent of the rest of the graph, given its neighbours in the graph and the observations.

Furthermore, the distribution of a CRF must factorise according to the cliques of the graph; cliques are fully connected (or complete [12]) sub-graphs, i.e., there exists an edge between all pairs of nodes in the sub-graphs. Let C be the set of all cliques of the graph. The distribution must then factorise as a product of clique potentials  $\phi_c(x_c, \mathbf{z})$  as follows:

$$p(\mathbf{x} \mid \mathbf{z}) = \frac{1}{Z(\mathbf{z})} \prod_{c \in \mathcal{C}} \phi_c(x_c, \mathbf{z}), \tag{4}$$

where  $x_c$  are the hidden variables of the clique and  $Z(\mathbf{z}) = \sum_x \prod_{c \in \mathcal{C}} \phi_c(x_c, \mathbf{z})$  is the partition function; it normalises the exponential making it into a proper distribution.

Clique potentials map variable configurations to non-negative numbers and capture the compatibility among the variables in the clique. They are commonly expressed by a log-linear combination of feature functions,  $\phi_c(x_c, \mathbf{z}) = \exp(\mathbf{w}_c^T \cdot \mathbf{f}_c(x_c, \mathbf{z}))$ , resulting in the following probabilistic definition of a CRF:

$$p(\mathbf{x} \mid \mathbf{z}) = \frac{1}{Z(\mathbf{z})} \exp\left(\sum_{c \in \mathcal{C}} \mathbf{w}_c^T \cdot \mathbf{f}_c(\mathbf{x}_c, \mathbf{z})\right);$$
(5)

where  $Z(\mathbf{z}) = \sum_{x} \exp(\sum_{c \in \mathcal{C}} \mathbf{w}_{c}^{T} \cdot \mathbf{f}_{c}(x_{c}, \mathbf{z}))$  is the partition function expressed using the log-linear form. *C* is again the set of all cliques in the graph. The feature functions  $\mathbf{f}_{c}$  extract feature vectors given the value of the clique variables  $x_{c}$  and observations  $\mathbf{z}$ .  $\mathbf{w}_{c}$  are parameters (or weights) which express the relative importance of the feature functions  $\mathbf{f}_{c}$  and will be estimated by performing learning.

The conditional distribution  $p(\mathbf{x}|\mathbf{z})$ , usually the distribution of interest, is directly modelled by a CRF. This is in contrast to generative models such as naive Bayesian models, Hidden Markov Models and Markov Random Fields. Generative models rely on modelling the intractable  $p(\mathbf{Z})$  and application of Bayes' rule to infer hidden states. The intuition is that a CRF can be used to model variables which are spatially or temporally correlated according to the structure of the underlying graph. As a result, CRFs have become a popular modelling and classification technique (see [13–15]) since they were first developed for the purpose of labelling sequence data [10].



**Fig. 3.** A chain graph CRF is employed to model the laser scan matching problem.  $\{L_{ref}^i\}_{i=1}^N$  denotes the *N* points in the reference scan which are mapped to the *N* CRF latent variables.  $\{L_{new}^i\}_{i=1}^M$ , the *M* points of the new scan, are viewed as the observation set *z*.  $\phi_l(x_i)$  denotes the local potential, while  $\phi_p(x_i, x_j)$  represents the pairwise potential.

#### 3.2. From laser scans to conditional random fields

In this work we consider a particular type of CRF referred to as a pairwise CRF, it only contains two types of potential functions: local potentials and pairwise potentials. The laser scan matching problem is modelled by a chain CRF graph. The motivation for using a chain stems from the way scan data is obtained. The laser scanner measures range data in a single plane; the plane in which the beam scans. As such, laser points are acquired one after the other; a chain represents this acquisition. Correlations in the data are not lost in the graph as there is a path from any one node in the chain to any other node. Fig. 3 interprets how a chain CRF models the reference and new laser scans.

The laser point association problem is expressed by the conditional distribution  $p(\mathbf{x}|\mathbf{z})$ . The hidden variables  $\mathbf{x}$  represent the associations given an observation set  $\mathbf{z}$ . The observation set consists of the laser sensor measurements  $L_{\text{ref}}$  and  $L_{\text{new}}$ .

As shown in Fig. 3, the hidden variables **x** are represented by *N* nodes. Each of these nodes is discrete with M + 1 states. Here *N* and *M* are as they have been defined in Section 2. The states of a node, say node  $x_i$ , have the following interpretation. The first state indicates the likelihood that laser point  $L_{\text{ref}}^i$  in the reference scan associates to the first laser point  $L_{\text{new}}^i$  in the new scan. The second state is the likelihood of association to the second point in the new scan, etc. In this way, node  $x_i$  can range over all points in the new laser scan. Finally, the (M + 1)th state represents the likelihood that  $L_{\text{ref}}^i$  is an outlier.

For the scan matching problem the general formulation of a CRF, Eq. (4), can be re-written into:

$$p(\mathbf{x} \mid \mathbf{z}) = \frac{1}{Z(\mathbf{z})} \prod_{i \in \mathcal{V}} \phi_l(x_i, \mathbf{z}) \prod_{(i,j) \in \mathcal{E}} \phi_p(x_i, x_j, \mathbf{z}),$$
(6)

here, the term  $Z(\mathbf{z})$  refers again to the partition function. The local clique potential  $\phi_l(x_i, \mathbf{z})$  represents what a node *thinks* its state is, given the observations. In our case, they are vectors with (M + 1) elements when evaluating the potential for all  $x_i$  values. The *j*th entry of  $\phi_l(x_i, \mathbf{z})$  expresses the likelihood that  $L_{\text{ref}}^i$  associates to  $L_{\text{new}}^j$ . The pairwise potential  $\phi_p(x_i, x_j, \mathbf{z})$  guarantees consistency between the individual nodes' associations. Each pairwise potential relates nodes  $x_i$  and  $x_j$  on either end of an edge. They transform the state of node  $x_j$  into something node  $x_i$  understands and vice versa. They are represented by an  $(M + 1) \times (M + 1)$  matrix when evaluating all possible values of  $x_i$  and  $x_j$ . Using the log-linear form, the two types of potentials can be formulated as follows [16]:

$$\phi_l(x_i, \mathbf{z}) = \exp\left(\mathbf{w}_l^l \cdot \mathbf{f}_l(x_i, \mathbf{z})\right),\tag{7}$$

$$\phi_p(x_i, x_j, \mathbf{z}) = \exp\left(\mathbf{w}_p^T \cdot \mathbf{f}_p(x_i, x_j, \mathbf{z})\right)$$
(8)

where  $\mathbf{f}_l$  and  $\mathbf{f}_p$  denote the local and pairwise feature functions respectively, which are calculated from the range measurements. Section 3.3 will address the feature functions in more detail.  $\mathbf{w}_l$  and  $\mathbf{w}_p$  are their corresponding weights, these are estimated during the learning phase which is discussed in Section 3.4.

#### 3.3. Feature modelling

As shown above, the feature functions are the means by which the mapping from raw consecutive scans to the probabilistic framework of a CRF graph is achieved. They are discussed next.

#### 3.3.1. Association local features

The local features for association can be categorised into two classes. The first class of local features are boosting features including Data Boosting and Outlier Boosting. We construct these two features in the following manner. First, some basic geometric *features* are calculated from the underlying geometric properties of the raw measurements. The features identify patterns (shapes) around a single point in the reference scan. These same patterns are computed around *each* point in the new scan. The resulting difference between a pattern in the reference scan and all patterns in the new scan is an error metric. Points in the new scan that have a similar pattern (a small value for the error metric) are more likely candidates for association to the point in the reference scan. Next. these basic geometric features are used as inputs to boosting [17]. The outputs of boosting (the experiments employ AdaBoost [18] whose parameters are learned from pre-training) are used as local feature values. Using boosting allows the local features to better deal with non-linearities in the data; resulting in better estimates for the local potentials. Fig. 4 describes the construction of boosting feature values.

Four types of basic geometric features are employed in this work, they are formulated as follows:

• Distance:

$$\mathbf{f}_{dis}^{k}(i,j) = \left\| \|L_{ref}^{i} - L_{ref}^{i+k}\| - \|L_{new}^{j} - L_{new}^{j+k}\| \right\|.$$
(9)

• Angular:

$$\mathbf{f}_{\mathrm{ang}}^{k}(i,j) = \left\| \angle (\overline{L_{\mathrm{ref}}^{i-k}L_{\mathrm{ref}}^{i}}, \overline{L_{\mathrm{ref}}^{i}L_{\mathrm{ref}}^{i+k}}) - \angle (\overline{L_{\mathrm{new}}^{j-k}L_{\mathrm{new}}^{j}}, \overline{L_{\mathrm{new}}^{j}L_{\mathrm{new}}^{j+k}}) \right\|.$$
(10)

- Geodesic:
- $\mathbf{f}_{geo}^k(i,j)$

$$= \left\| \sum_{h=i}^{i+k-1} \|L_{\text{ref}}^{h} - L_{\text{ref}}^{h+1}\| - \sum_{h=j}^{j+k-1} \|L_{\text{new}}^{h} - L_{\text{new}}^{h+1}\| \right\|.$$
(11)

• Radius:

$$\mathbf{f}_{\text{rad}}(i,j) = \left\| \|\mathbf{O}_{\text{ref}} - L_{\text{ref}}^{i}\| - \|\mathbf{O}_{\text{new}} - L_{\text{new}}^{j}\| \right\|.$$
(12)

Here, the || || denotes the Euclidean distance.  $k \in \{-5, -3, -1, 1, 3, 5\}$  are the laser point index offsets. Thus distance and geodesic features will be computed six times while the angular feature is computed 3 times to account for various *k* value. Therefore, a total of 16 basic geometric features are extracted and used as input to AdaBoost. Note that using different offsets allows the CRF to deal with different scan point densities. For example, an object measured from a far away distance will consist of only a few points. That same object measured nearby will consist of many more points. Using different offsets allows the shape of the object to be matched regardless of the distance to the sensor. In Eq. (12),  $O_{ref}$  and  $O_{new}$  denote the original of the sensor-centric frames at the reference and new robot poses respectively.



**Fig. 4.** The construction of boosting feature values. Four types of basic geometric features are used as the inputs to AdaBoost. The outcome are boosting features consisting of data boosting and outlier boosting. For all the feature vectors, darker shades imply larger magnitude. In this example, we assume the new scan consists of three points.

Computation of the above four basic geometric features, for each *i*, results in a *M*-element vector in which each element implies the likelihoods of associations between a given reference point and all points of new scan with respect to the geometric property. These vectors are directly pushed into AdaBoost. The resulting outcome, which has the same size as the input vectors, are considered the Data Boosting feature values. This procedure is visualised by the blue arrow in Fig. 4. The *minimums* of these basic geometric features represent the most likelihood association. Classification of them by AdaBoost results in the Outlier Boosting features, which is scalar. The green arrows in Fig. 4 show the construction of Outlier Boosting feature values.

The second class of local features utilises the structure of the data. This class of features has only one member; the ICP association feature. It is advantageous to integrate ICP-based laser scan matching, provided the relative motion is not very significant. Furthermore, it facilitates inclusion of dead reckoning data if odometry is available. The framework of the ICP association feature is shown in Algorithm 1.

The inputs to the algorithm,  $\mathbf{L}_{ref}$  and  $\mathbf{L}_{new}$ , are the two consecutive laser scans.  $\textbf{RT}_{\text{odo}}$  is the initial rotation and translation from the odometry. It can be arbitrarily initialised if odometry data is unavailable.  $n_{ite}$  is the iteration number.  $\delta$  denotes the motion threshold which can terminate the iterative nearest neighbour search. In essence, the ICP algorithm is performed over the reference and new laser scans. First, the best association configuration for all the reference points is computed, see Lines 2 - 10. The function Trans*formFrame* performs frame compounding, and  $L_{new \rightarrow ref}$  denotes the new scan viewed from the reference frame. The function Nearest-Neighbours estimates the associations from the nearest Euclidean distance metric. The function *ComputeRT* is the implementation of [2, Appendix C]. Second, a mapping is required from the singlevalued ICP association results for each reference laser point  $L_{ref}^{i}$ (i.e., the node  $x_i$  of the chain CRF graph) to the multi-valued representation of the node's state. This is performed on Lines 11 - 15. The mapping is achieved by assigning 1 to the associated state  $C_{nn}[i]$  of node  $x_i$ , and zero to all its other states; where  $C_{nn}[i]$  is determined by the ICP association for node  $x_i$  (or the reference point  $L_{ref}^i$ ). As a result, the ICP association feature  $\mathbf{f}_{ICP}^{x_i}$  for node  $x_i$  is obtained.

Al	gorithm	1 ICP	association	feature	calcu	lation
----	---------	-------	-------------	---------	-------	--------

1: **Input**:  $\mathbf{L}_{ref}$ ,  $\mathbf{L}_{new}$ ,  $\mathbf{RT}_{odo}$ ,  $n_{ite}$ ,  $\delta$ 2:  $\hat{\mathbf{RT}} \leftarrow \mathbf{RT}_{odo}$ 3: while  $n_{\text{ite}} > 0$  and  $\Delta \mathbf{RT} > \delta$  do  $\mathbf{L}_{\text{new} \rightarrow \text{ref}} \leftarrow \text{TransformFrame}(\mathbf{L}_{\text{new}}, \hat{\mathbf{RT}})$ 4  $\mathbf{C}_{nn} \leftarrow \text{NearestNeighbours}(\mathbf{L}_{new \rightarrow ref}, \mathbf{L}_{ref})$ 5.  $\check{\mathbf{RT}} \leftarrow \text{ComputeRT}(\mathbf{C}_{nn}, \mathbf{L}_{ref}, \mathbf{L}_{new})$ 6:  $\Delta \mathbf{RT} \leftarrow \|\mathbf{RT} - \mathbf{RT}\|$ 7:  $\hat{\mathbf{RT}} \leftarrow \check{\mathbf{RT}}$ 8: 9:  $n_{\text{ite}} \leftarrow n_{\text{ite}} - 1$ 10: **end while** 11: **for** i = 1 to *N* **do**  $\mathbf{f}_{\text{ICP}}^{x_i} \leftarrow \mathbf{0}_{1 \times (M+1)}$  $j \leftarrow \mathbf{C}_{\text{nn}}[i]$  $\mathbf{f}_{\text{ICP}}^{x_i}[j] \leftarrow 1$ 12: 13: 14: 15: end for 16: **Return \mathbf{f}\_{\text{ICP}}^{x\_1}, \mathbf{f}\_{\text{ICP}}^{x\_2}, ..., \mathbf{f}\_{\text{ICP}}^{x\_N}** 

The ICP association feature will ensure that a significant amount of probability mass is located on likely associations. The shape based boosting features ensure sufficient probability mass is spread over the remaining states to allow the inference algorithm to solve for a solution that is consistent over the graph.

The data boosting, outlier boosting and ICP association features are multiplied by their individual weights, before being combined to construct the (M + 1)-element local potential.

#### 3.3.2. Association pairwise features

These features also fall into two categories. Those that use the structure of the scan acquisition to relate the state of nodes, and those that use the observations.

The first category of features use the scan acquisition structure. In an ideal world, without noisy measurement and outliers, it would be straight forward to relate the association of node  $x_i$  with that of node  $x_{i+1}$ . If node  $x_i$  associates to point  $L_{new}^j$  then node  $x_{i+1}$ can reasonably be expected to associate to point  $L_{new}^{j+1}$ .

- Sequence: Expresses the above sequential nature of association by an identity matrix with the diagonal shifted up. Because of outliers, there will occasionally be jumps in the sequence. This can be accommodated by a sequence feature where the diagonal is shifted upwards by an offset greater than one. In the experiments we add features with offsets of 1 7.
- Pairwise outlier: Expresses how outliers impact on association transitions – from inlier to outlier and vice versa.

The second category of pairwise features operate similarly to the basic geometric features. These use a measure between two points corresponding to an edge in the reference scan and compare this measure with *all* possible combinations of this measure in the new scan. The comparison produces a metric of how pairs of points are related between the two scans — i.e. a transition. Here only one such feature is used.

 Pairwise distance: Uses the distance between points on either end of an edge.

#### 3.4. Maximum pseudo-likelihood learning

Learning the values of the weights  $\mathbf{w}_c$  is achieved through maximisation of the conditional likelihood (Eq. (5) or (6)) given labelled training data. In our case this is computationally intractable; the partition function  $Z(\mathbf{z})$  sums over the (very large) space of all hidden variables. We therefore employ maximum pseudo-likelihood learning [19].

Maximum pseudo-likelihood (MPL) learning approximates the joint distribution by considering, for each hidden variable, only its immediate neighbours in the graph, i.e., the nodes contained in its Markov blanket. As a result, computing the partition function is simplified significantly by only requiring summation over the states of a single hidden variable at a time. Specifically, MPL learning maximises the following *pseudo-likelihood*:

$$pl(\mathbf{x} \mid \mathbf{z}, \mathbf{w}) \triangleq \prod_{i=1}^{N} p(x_i \mid MB(x_i), \mathbf{w})$$
$$= \prod_{i=1}^{N} \frac{\exp\left\{\mathbf{w}_c^T \cdot \mathbf{f}(x_i, MB(x_i), \mathbf{z})\right\}}{Z(MB(x_i), \mathbf{w})},$$
(13)

where *N* is the number of nodes in the graph, MB( $x_i$ ) is the Markov blanket of the hidden variable  $x_i$ ,  $p(x_i | MB(x_i), \mathbf{w})$  is the *local like-lihood* of  $x_i$ ,  $\mathbf{f}(x_i, MB(x_i), \mathbf{z})$  denotes the local feature value involving variable  $x_i$  and  $Z(MB(x_i), \mathbf{w}) = \sum_{x'_i} \exp \{\mathbf{w}_c^T \cdot \mathbf{f}(x'_i, MB(x'_i), \mathbf{z})\}$  represents the local normalising function. In this way, computing the expensive global partition function is avoided. Even though MPL learning is an approximation, it has been shown to give good results in various domains; see [20,21,8].

For practical reasons optimising the *pseudo-likelihood* is performed on the negative of its log, resulting in the following cost function:

$$\mathcal{PL}(\mathbf{w}) = -\sum_{i=1}^{N} \log p(x_i \mid \text{MB}(x_i), \mathbf{w}) + \frac{(\mathbf{w} - \widetilde{\mathbf{w}})^T (\mathbf{w} - \widetilde{\mathbf{w}})}{2\sigma^2}.$$
(14)

The second term on the right is a regulariser, or shrinkage prior. It prevents solutions for  $\mathbf{w}$  in the extreme while at the same time allowing a Gaussian prior over the weights. In the experiments a uniform zero mean prior is used. Thus Eq. (14) becomes

$$\mathcal{PL}(\mathbf{w}) = \sum_{i=1}^{N} \left( -\exp\left\{\mathbf{w}_{c}^{T} \cdot \mathbf{f}(x_{i}, \mathrm{MB}(x_{i}), \mathbf{z})\right\} + \log Z(\mathrm{MB}(x_{i}), \mathbf{w}) \right) + \frac{\mathbf{w}^{T}\mathbf{w}}{2\sigma^{2}}.$$
(15)

We then apply gradient-based algorithms to find an optimal **w** by minimising  $\mathcal{PL}(\mathbf{w})$ .

#### 3.5. Max-sum belief propagation inference

A CRF (Eq. (5) or (6)) defines a distribution over all hidden variables. Typically one is more interested in the distribution over a single hidden variable (the marginal), or the configuration for which that variable achieves its maximum *a-posteriori* (MAP) value. Inference is the process by which these quantities can be obtained. The Belief Propagation [22] algorithm is able to perform inference efficiently. Two variants of belief propagation are widely used, Sum-Product and Max-Product [23]. In the laser scan matching case, the former can estimate the marginal distributions for association of every laser point in the reference scan to each point in the new scan (and the outlier state). The latter allows us to find the most likely configuration, the associated laser point pairs between the two scans, and to find its MAP value. In practice, we wish to find the set of laser point associations that jointly have the largest likelihood. Therefore, the Max-Sum inference procedure (Max-Product in the log domain) is adopted in this paper.

Belief Propagation (BP) is a class of inference algorithms in which each node sends messages to each of its neighbours in the graph. The messages, which are essentially distributions (not necessarily normalised), convey what a node believes its neighbours' state should be given its own state. With respect to Max-Product Belief Propagation, the received messages together with a node's own belief are then used to compute the MAP configuration for which the joint achieves its maximum. The messages are defined as follows:  $m_{ii}^{\max}(x_i)$ 

$$m_{ij}^{\max}(x_j) = \max_{x_i} \left( \phi_l(x_i) \phi_p(x_i, x_j) \prod_{k \in \mathcal{N}(i) \setminus j} m_{ki}^{\max}(x_i) \right).$$
(16)

Here  $m_{ii}^{\max}(x_i)$  is the message node  $x_i$  sends to node  $x_i$ ; i.e. what node  $x_i$  thinks node  $x_i$ 's state should be.  $\phi_l(x_i)$  and  $\phi_p(x_i, x_i)$ represent the local and pairwise potential values respectively (see Section 3.2).  $\mathcal{N}(i) \setminus j$  denotes node  $x_i$ 's neighbours other than node  $x_i$ . The constructions of the messages make use of the distributive law of the max operator. So max and products are interchanged for the sake of efficiency. In case of a CRF with a tree structure, the messages are passed from leave nodes towards an arbitrarilydefined root node. The MAP solution can be obtained by invoking the following equation at the root node  $x_r$ ,

$$p^{\max}(\mathbf{x} \mid \mathbf{z}) \propto \max_{x_r} \left( \phi_l(x_r) \prod_{j \in \mathcal{N}(r)} m_{jr}^{\max}(x_r) \right).$$
(17)

However, a practical problem is that the product of small probabilities tends to numerically underflow. This can be handled by working with the logarithm of the joint distribution. The message construction indicated in Eq. (16) then becomes

$$= \max_{x_i} \left( \ln \phi_l(x_i) + \ln \phi_p(x_i, x_j) + \sum_{k \in \mathcal{N}(i) \setminus j} m_{ki}^{\max}(x_i) \right).$$
(18)

It is of interest to note that, since a chain CRF is employed in this paper, each node possesses no more than two neighbours. That is, Eq. (18) can be simplified to

$$m_{ij}^{\max}(x_j) = \max_{x_i} \left( \ln \phi_l(x_i) + \ln \phi_p(x_i, x_j) + m_{ki}^{\max}(x_i) \right),$$
(19)

where node  $x_k$  is the other neighbour of node  $x_i$  besides node  $x_i$ . Analogously, the MAP solution becomes

$$p^{\max}(\mathbf{x} \mid \mathbf{z}) \propto \max_{x_r} \left( \phi_l(x_r) + m_{jr}^{\max}(x_r) \right).$$
<sup>(20)</sup>

The key point of CRF-Matching is to find the state of each node which contributes to the MAP solution. These can be viewed as the laser point correspondences between the new laser scan and the reference scan. This can be achieved by computing the MAP configuration as discussed next. On the message pass from leaves to root, with each message  $m_{ii}$  sent from node  $x_i$  to its parent node  $x_i$ , the node also records the maximising states of  $x_i$  responsible for  $m_{ii}$ . The states are recorded in a state table as follows,

$$\delta_{ij}(x_j) \in \arg \max_{x_i} \left( \ln \phi_l(x_j) + \ln \phi_p(x_i, x_j) + m_{ki}^{\max}(x_i) \right).$$
 (21)

Once at the root  $x_r$ , Eq. (19) becomes

$$\delta_{jr}(x_r) \in \arg\max_{x_j} \left( \ln \phi_l(x_r) + \ln m_{jr}(x_r) \right).$$
(22)

Furthermore, the MAP state of the root is computed as

$$x_r^* \in \arg\max_{x_r} \left( \ln \phi_l(x_r) + \ln m_{jr}(x_r) \right)$$
(23)

where  $x_r^*$  is the maximal configuration at the root. As can be seen, maximisation is performed on the root's own belief (local potential) combined with all incoming messages. Again, Eq. (23) is for the chain CRF case. The maximal configuration for the root node is then simply a state for which the combined belief is maximal. Using this maximal configuration of the root, the algorithm backtracks along the graph using  $\delta_{ij}$ . For example, suppose that as is shown in Eq. (22), node  $x_j$  is the immediate child of the root node, then node  $x_j$ 's maximal configuration can be extracted as  $x_j^* = \delta_{jr}(x_r^*)$ . A consistent maximal configuration  $\mathbf{x}^*$  for all nodes can be found by backtracking from root to leave nodes based on the recorded  $s_{ii}$ . As a result,  $\mathbf{x}^*$  are the laser point indexes (the outlier state is accounted for as well) in the new laser scan which associates with the points of the reference scan.

Algorithm 2 (Sampling-based uncertainty estimation) 1: **Input**: *K*, *G*, **L**<sub>ref</sub>, **L**<sub>new</sub> 2: Initialisation: Choose root node  $x_r$  and leaf node  $x_l$ Choose *inward* ordering  $I_{in}$  and *outward* ordering  $I_{out}$ 3: **for** k = 1 to *K* **do** 4: for all  $\langle x_i, x_i \rangle \in \mathcal{I}_{in}$  do 5: **if** *x*<sub>*i*</sub> is leaf **then**  $\Theta_{ii} \leftarrow \phi_l(x_i)\phi_p(x_i, x_i)$ 6: else 7:  $\Theta_{ij} \leftarrow \phi_l(x_i)\phi_p(x_i, x_j) \prod_{k \in \mathcal{N}(i) \setminus j} m_{ki}^{\mathrm{sam}}(x_i)$ 8: 9: end if **for** s = 1 to M + 1 **do** 10:  $(m_{ii}^{\text{sam}}[s], \delta_{ii}^{\text{sam}}[s]) = \text{CPS}(\Theta_{ij}[s, :])$ 11: 12: end for end for 13: At root  $x_r$ :  $\begin{array}{l} \theta_r \leftarrow \phi_l(x_r) \prod_{j \in \mathcal{N}(r)} m_{jr}^{\text{sam}}(x_r) \\ (\mathcal{L}_r, \ c_r) = \text{CPS}\left(\theta_r\right) \end{array}$ 14: 15: for all  $\langle x_j, x_i \rangle \in \mathcal{I}_{out}$  do 16:  $c_i = \mathscr{S}_{ij}^{\rm sam}[c_j]$ 17: 18: end for 19:  $\mathbf{C} \leftarrow \begin{bmatrix} c_r, \ldots, c_i, c_j, \ldots, c_l \end{bmatrix}$  $(\mathbf{R}[k], \mathbf{T}_{x}[k], \mathbf{T}_{y}[k]) = \text{ComputeRT}(\mathbf{C}, \mathbf{L}_{\text{ref}}, \mathbf{L}_{\text{new}})$ 20: 21: end for 22:  $\Sigma_{\mathbf{RT}} = \text{ComputeCovariance}(\mathbf{R}, \mathbf{T}_{x}, \mathbf{T}_{y})$ 23: **Return**  $\Sigma_{RT}$ 4. Sampling inference based uncertainty estimate

The uncertainty of laser scan matching in the context of robotics generally refers to the relative motion estimated uncertainty. As addressed in Section 1, the uncertainty from improper laser point associations plays a much more crucial role than that from sensor noise. In order to estimate the uncertainty, several association hypotheses are required. In [8] it was shown that Max-Sum inference generates optimal results for CRF scan matching. Intuitively this makes sense as the MAP configuration maximises the joint distribution rather than the marginal for each node. For our motivation, we therefore wish to sample configurations that map to likely relative motion estimates. This means that it is not possible to randomly sample configurations from the association joint distribution as presented in [24]; an average pose and not a maximal pose is the expected result. Likewise, it is not possible to find the *K* most probable configurations as shown in [25]; there is no guarantee that this will result in relative motion near that of the MAP configuration.

#### 4.1. Algorithm outline

For the remainder of this section we shall focus on establishing an efficient approach to fitting the uncertainty distribution of a robot's relative motion estimates based on our CRF scan matching framework. To accomplish this task we assume that the mapping from association to pose is smooth; the uncertainty of relative motion follows from a Gaussian distribution and in the experiments a least squares mapping is used. With these assumptions, the problem is intuitively one of sampling configurations that are similar to the MAP configuration. We achieve this with the CRF inference procedure presented in Section 3.5. The configuration sampling should take account of the different likelihoods<sup>1</sup> with which each node associates to. More precisely, we need to track the underlying probability within message propagation while performing inference.

<sup>&</sup>lt;sup>1</sup> It could be viewed as probability, after being normalised.



1: **Input**: a vector **a** with *n* elements 2:  $\mathbf{b} \leftarrow \mathbf{a} / \sum_{i=1}^{r} \mathbf{a}[i]$ 3: **for** i = 1 to *n* **do** 4:  $\mathbf{d}[i] = \sum_{j=1}^{i} \mathbf{b}[j]$ 5: **end for** 6:  $r \leftarrow \text{GenerateRandomNumber}([0, 1])$ 7:  $t \leftarrow \arg\min_{k} (\mathbf{d}[k] \ge r)$ 8:  $v \leftarrow \mathbf{a}[t]$ 9: **Return** v, t

As an implementation of the idea, a sampling-based uncertainty estimate algorithm is proposed in this paper. Its outline is shown in Algorithm 2.

The inputs to the proposed algorithm are as follows:

- The coordinates of the laser points in the reference scan L<sub>ref</sub> and new scan L<sub>new</sub>, which we have already defined in Section 2.
- The CRF graph *G* that we established in Section 3.2. Although the chain CRF is employed in this paper, the proposed algorithm can work with a tree graph as well.
- The number *K* of the association hypothesis that we wish to generate. Each association hypothesis corresponds to a relative motion solution  $[R, T_x, T_y]$ , i.e., the rotation, the translation on *x* and *y*.

The covariances of rotation and translation, denoted as  $\Sigma_{RT}$ , are returned from Algorithm 2.

#### 4.2. Sampling-Product message construction

Analogous to the applications in [24,25], our approach is based on sampling techniques. However, as shown in Algorithm 2, we perform the sampling along with message construction. That is, we seek a new way to construct messages during inference which can generate a set of probable association configurations. Recall the message construction addressed in Section 3.5, we explicitly eliminate unwanted variables via the max operator when processing the probabilistic information from the pairwise potential, the local potential and incoming messages. We therefore start the Sampling-Product through modification of Max-Product message construction. The pseudocode for message construction is presented from Line 4 to Line 13 of Algorithm 2.

We refer to an *inward* ordering  $\mathcal{I}_{in}$  in which messages are propagated from the leave nodes towards the root node, and reversely an outward ordering  $\boldsymbol{\mathit{1}}_{out}$  in which messages flow from the root towards the leaves.  $\langle x_i, x_j \rangle \in \mathfrak{I}_{\mathrm{in}}$  represents two adjacent nodes among which a message is sent from  $x_i$  to  $x_j$ .  $\Theta_{ij}$  is the product of local potential, pairwise potential and all received messages. Note that at the leaf nodes, the message can be considered as **1** (see Line 6).  $\Theta_{ij}$  is a likelihood table indicating the joint likelihood of the association configurations of  $x_i$  and  $x_i$ . Fig. 5 offers a graphical illustration of the Sample-Product message construction procedure. That is, instead of selecting the state with maximal likelihood volume along each row of  $\Theta_{ii}$ , which Max-Product does, we view each row as a Sum of Gaussian (SoG) [26] and pick up a state by Cumulative Probability Sampling (CPS) shown as Algorithm 3. These selected states constitute the message.

The CPS is similar to the general resampling technique widely used in Monte Carlo filtering [27]. It guarantees that the randomly selected state reflects its original likelihood and maintains the feasibility as well. Assuming a vector **a** has *n* elements, each corresponding to the likelihood of one potential state, we can select a state probabilistically by performing CPS. Firstly, *a* is normalised as Line 2 of Algorithm 3 does. Then the cumulative probability is



**Fig. 5.** A graphical explanation of Sample-Product message construction. (top) Node  $x_i$  is propagating  $\Theta_{ij}$  to node  $x_j$  along  $I_{in}$  of a chain CRF graph (see Algorithm 2). (bottom) On the left is the visualisation of  $\Theta_{ij}$ . Darker shades in the matrix imply larger magnitude, here we assume the new scan has three points, then  $\Theta_{ij}$  is a  $4 \times 4$  matrix. Subsequently, we consider each row of  $\Theta_{ij}$  as a Sum of Gaussian (SoG) distribution and perform sampling to select a state. The procedure is shown on the right.



**Fig. 6.** An example of Cumulative Probability Sampling (CPS) (shown in Algorithm 3). (left) A vector with four states [0.1, 0.2, 0.4, 0.3]. (middle) The visualisation of the probability volume for each state. (right) Cumulative probability function is computed, a random number r = 0.55 is generated, and then the third state with the probability of 0.4 is selected.

computed for each element a[i], meanwhile a random number r is generated over [0, 1]. Finally, the state and its likelihood are selected according to the equation listed on Lines 7 and 8. CPS guarantees that the state with highest probability is most likely selected. As illustrated in a CPS example shown in Fig. 6, the state with the probability of 0.4 is most probably sampled, because it contributes most to the cumulative probability function.

#### 4.3. Potential configuration backtracking

Backtracking in the proposed algorithm parallels that of Max-Product (or Max-Sum) inference (see Section 3.5). As shown on Line 11 of Algorithm 2, both the selected state's likelihood and its index are returned by CPS. The latter is recorded in the sampled state table  $\delta_{ij}^{sam}$  for each node when propagating the sampled messages in the *inward* order  $\ell_{in}$ . When the messages arrive at the root node  $x_r$ , a special version of  $\Theta$ , denoted as  $\theta_r$  (see Line 14 of Algorithm 2), is constructed from the local potential  $\phi(x_r)$  and all incoming messages. Again, we consider it as a SoG distribution and perform CPS over the column vector (see Lines 14 and 15 of Algorithm 2). The selected state at the root node is used for the configuration backtracking. This is encoded from Line 16 to 18 of Algorithm 2 and graphically represented by Fig. 7. The *outward* backtracking generates a configuration based on  $\delta^{sam}$ .



**Fig. 7.** A graphical illustration of backtracking in Sampling-Product inference (top). Root node  $x_r$  and its adjacent nodes  $x_j$  and  $x_i$  along  $I_{in}$  in a chain CRF graph (see Algorithm 2) (bottom). Backtracking from  $x_r$  to  $x_i$ . First,  $\theta_r$  is the product of  $x_r$ 's local potential and the message from  $x_j$ , where we again assume that the new laser scan consists of three points. Next, CPS is carried out over  $\theta_r$ , and assume state 1 is selected. Note that state 0 corresponds to the *outlier* state. Lastly, we find state 3 of  $x_j$ , which contributed to state 1 of  $x_r$  by checking the first entry of  $\vartheta_{jr}^{sam}$ . This process is repeated for each node, the third entry of  $\vartheta_{ij}^{sam}$  indicates the potential configuration of node  $x_i$ .

After finding the sampled state c for all the nodes in the CRF graph a potential configuration C, which represents the laser point association hypothesis between the reference and new scan, is obtained.

#### 4.4. Relative motion uncertainty estimate

As illustrated on Lines 3 – 21 of Algorithm 2, *K* association hypotheses are generated by running the *for* loop *K* times. On Line 20, each *C* is transformed into a solution to the rotation *R* and translation  $T = [T_x, T_y]$  according to the method used by ICP [2]. It is formulated as

$$R, T \leftarrow \arg\min_{R,T} \sum_{i=1}^{N} \left\| R \mathbf{L}_{new}^{i} + T - \mathbf{L}_{ref}^{c_i} \right\|,$$
(24)

where N denotes the number of laser points in the reference scan.  $c_i$  is the index of the point in the new scan, associated to the *i*-th point in the reference scan. As such, association hypothesis are mapped to points in the robot relative motion space.

*K* potential solutions to the relative motion are obtained, each with a Gaussian distribution assumption. The uncertainty is computed as shown on Line 22 of Algorithm 2. We define **RT** =  $[\mathbf{R}, \mathbf{T}_x, \mathbf{T}_y]$ . The covariance of **RT** can be calculated as follows:

$$\Sigma_{\mathbf{RT}} = E\left((\mathbf{RT} - E(\mathbf{RT}))(\mathbf{RT} - E(\mathbf{RT}))\right),\tag{25}$$

where *E* denotes the expectation operator. In Section 5 we shall show how  $\Sigma_{RT}$  can be integrated into the filtering framework of a SLAM application.

#### 4.5. Computational costs

Instead of ranging over the  $\mathcal{O}((M + 1)^N)$  possible associations to compute the uncertainty, the proposed algorithm tracks the underlying uncertainty by focusing on the *K* probable laser point associations. The computational expense relates to the number of particles that we draw. In terms of the implementation, an objectoriented CRF toolbox, which includes the proposed Sampling-Product inference procedure, has been developed at the Australian Centre for Field Robotics (ACFR). It is based on MATLAB 2010a and takes advantage of its object-oriented framework. The features



Fig. 8. The processing time for drawing different numbers of samples.

addressed in Section 3.3 are integrated as classes. To determine the efficiency, we estimate the uncertainty of matching a laser scan pair by running the ACFR-CRF toolbox on a low-cost laptop. The resulting processing time, for different numbers of particles, are shown in Fig. 8. Here, 100 valid laser points are included for each of the scans. In the experiments we use 25 samples which takes roughly 2.3 s. This set up represents a good tradeoff between computational cost and performance.

The ACFR-CRF toolbox supports distributed computing for use on multi-core CPU or multi-computer systems. It allows the inference procedures to be performed over multilaser scan pairs simultaneously thus saving on computational cost. With the distributed computing technique, the matching and uncertainty quantification for 4391 laser scan pairs, as used in Section 5.2, are processed within 15 min.

#### 5. Experiments

To verify whether the proposed algorithm is able to capture the underlying uncertainty of scan matching, we evaluate its performances on two data sets. The first data set consists of underconstrained simulated data in the form of corridor and circular environments. The second is a data set obtained at the *University of Sydney* campus and is used in a Simultaneous Localisation and Mapping (SLAM) task.

#### 5.1. Simulated data set

As mentioned in Section 1, there are two under-constrained environments; the corridor and circular environments. These lead to large uncertainty in specific directions when aligning the pairwise scans. The expected shapes of the uncertainty ellipsis allow us to check the validity of the proposed algorithm [6,7].

#### 5.1.1. Corridor environment

In the simulated corridor environment the robot is located in a long corridor. The length of the corridor extends beyond the range of the range finder, the robot is thus only able to sense the walls.

Laser scans viewed from two consecutive poses, assuming the walls are sufficiently smooth, will appear identical to the robot. Therefore associated points are not necessarily the same physical points. As a result, a lack of distinguishing features for consecutive scans will significantly impair scan matching during inference in particular along the direction of the walls where most laser points





**Fig. 9.** Laser scan alignment uncertainty estimates for the circular environment. (a) Scenario; triangle indicates the robot. Yellow colour indicates the initial pose and range measurements. Green colour indicates the subsequent pose and measurements. (b) The X-Y uncertainty ellipsis and samples computed with Sampling-Product inference. (c) X-rotation uncertainty ellipsis and samples. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

are aligned. An extreme case is shown in Fig. 9(a); the robot moves in parallel with the walls. The laser range measurements are corrupted by Gaussian white noise with  $\mu_r = 0$ ,  $\delta_r = 0.06$  m.

The standard deviations of the relative motion estimates are [102.90 mm, 5.11 mm, 0.02°]. The  $3-\sigma$  ellipses shown in Fig. 9(b) and (c) demonstrate that the relative motion estimate on X is more uncertain than on Y.

#### 5.1.2. Circular environment

Analogous to the corridor case, when the robot stands in an ideal circular environment, the walls will significantly impair

**Fig. 10.** Laser scan alignment uncertainty estimates for the circular environment. (a) Scenario; triangle indicates the robot. Yellow colour indicates the initial pose and range measurements. Green colour indicates the subsequent pose and measurements. (b) The X-Y uncertainty ellipsis and samples computed with Sampling-Product inference. (c) X-rotation uncertainty ellipsis. (d) Y-rotation uncertainty ellipsis.

the robot's ability to correctly determine its rotation estimate. An extreme circular situation is pictured in Fig. 10(a). During inference, it is difficult for the CRF to find optimal laser point associations as there are strong similarities between all points in the data. However, compared to the corridor case, the uncertainties for laser point associations are expected to impact mainly on the rotation estimates. This is visualised by the long radius of the uncertainty ellipsis along the rotation axis as shown in Fig. 10(b) and (c). The standard deviation of the motion estimates for the circular environment are [94.30 mm, 70.20 mm, 2.35°]. The ground truth of uncertainty for improper laser point associations



Fig. 11. The utility vehicle (UTE) robot and SICK laser sensor employed for the laseronly SLAM implementation.

is difficult to obtain. However, it is still possible to draw an important conclusion from the results based on the shape of the uncertainty ellipsis. The results demonstrate that the proposed approach correctly models the underlying uncertainty in the under-constrained corridor and circular cases.

#### 5.2. Large-scale SLAM

The proposed sampling inference is also integrated into a SLAM application in a large-scale unstructured urban environment. The configuration of the mobile platform, a utility vehicle with SICK laser sensor, is shown in Fig. 11. The laser range finder data was acquired around the main campus of the University of Sydney during the day time, it therefore contains a significant number of dynamic objects, such as cars and people. To sidestep difficulties in feature extraction and retain computational tractability, a view-based delayed state SLAM framework is adopted. We do not explicitly model the environment by extracting features and estimating their positions: instead only a selection of key historical poses of the robot's trajectory are maintained in filtering. The raw laser scans are stored and registered in order to calculate pairwise constraints, detect potential loop closures and provide virtual observations of pose displacements [28, Chapter 4.2]. Furthermore, we utilise the extended information filter (EIF) rather than traditional extended Kalman filter (EKF) for practical reasons.

CRF laser scan matching functions, in some sense, can be seen as a *virtual odometry*. It can be used to generate estimates, based on consecutive laser scan alignments, for the control input  $\delta = [x_{\delta}, y_{\delta}, \theta_{\delta}]^{\top}$ . These are then incorporated into the robot motion model,

$$\bar{\mathbf{x}}(t) = \mathbf{x}(t-1) \oplus \delta(t) + \mathbf{w}(t).$$
(26)

Here  $\mathbf{x}(t - 1)$  is the estimate of robot pose at the previous time step, while  $\bar{\mathbf{x}}(t)$  corresponds to the pose prediction for the current time step.  $\oplus$  stands for the vector compounding operation [9, Section 3.2]. Process noise, due to the uncertainty of scan matching, is represented by  $\mathbf{w}(t)$ ; an error vector with a zero-mean Gaussian distribution with covariance  $\mathbf{Q}(t)$ .

The uncertainty covariance matrix  $\Sigma_{RT}(t)$ , computed from the hypothesis (see Section 4.4), is incorporated into the system as follows:

$$\mathbf{Q}(t) = B(t) \boldsymbol{\Sigma}_{\mathbf{RT}}(t) B(t)^{\top}$$
(27)

where B(t) is the Jacobian matrix of the motion model (Eq. (26)) with respect to the control input evaluated at the updated state of previous time step;

$$B(t) = \left. \frac{\partial(\mathbf{x} \oplus \boldsymbol{\delta})}{\partial \boldsymbol{\delta}} \right|_{\mathbf{x}(t-1)}.$$
(28)

The reader is referred to [28,29] for more details.

GPS data from the mobile platform was collected for groundtruth evaluation. For comparison, ICP-based SLAM with uncertainty estimates from the *offline* method [6,29] are computed. The trajectories and maps are visualised in Fig. 12 where the first loop closure corrects both the robot states and map representation.

Fig. 12(a) shows that the proposed CRF-SLAM method more closely follows the GPS ground truth.<sup>2</sup> Cumulative errors mean that, after an approximately 420 m displacement, ICP-SLAM starts to significantly deviate from the GPS ground truth. CRF-SLAM does not start to deviate until shortly before loop closure is detected. The final offsets in open-loop CRF-SLAM and ICP-SLAM are 24.8 m and 45.3 m respectively.

The corrected closed-loop trajectories of these two approaches show that CRF-SLAM outperforms ICP-SLAM according to GPS ground truth. The CRF trajectory is overlaid with  $1-\sigma$  uncertainty ellipsis for every 100 poses. The uncertainty is estimated by the delayed-state filter based on the covariance inferred from the algorithm discussed in this paper. As illustrated in Fig. 12(b), loop closure decreases the uncertainty significantly. The open-loop and closed-loop maps for ICP-SLAM and CRF-SLAM are also shown in Fig. 12.

The CRF-SLAM results presented here show that our algorithm to estimate the uncertainty of the motion can be seamlessly integrated into a SLAM framework with good performance.

#### 6. Discussions

Uncertainty estimation is indispensable within a filtering framework. The proposed Sampling-Product inference procedure incorporates CRF-Matching into practical SLAM applications. Existing work tackling this topic, such as [1,30,6,7,3,4], are based on Iterative Closest Point (ICP) and its variants. These are fundamentally different from laser scan matching with Conditional Random Fields (CRF). Ramos et al. [8] demonstrated that CRF-Matching outperforms ICP-based approaches. Furthermore, in contrast to these techniques, we focus on the uncertainty caused by improper laser point associations rather than measurement noises when performing scan matching. Thus, the proposed algorithm is not an evolution of existing uncertainty estimating algorithms but is exploiting the task in a new way, that is, capturing the laser point associations hypothesis with a shape-oriented graphical model.

The sampling approach has been employed to track the uncertainty distribution. The offline approach presented in [7] also addresses this problem. It attempts to model the huge laser point correspondence space  $\mathcal{O}((M + 1)^N)$  by random brute force association configuration test. It randomly selects a small set of particles and assigns an identical likelihood to each particle to estimate the uncertainty. However, the correspondence space is too large for a representative sample set. The offline approach cannot guarantee to always catch the underlying distribution. With respect to the mechanism of CRF-Matching, a single solution to rotation and translation can be found from the message constructions for the nodes within the CRF graph (see Section 3.5). For a single node, the combination of incoming messages, pairwise potential and local potential is the joint likelihood of association to the node. We view the likelihood as the Sum of Gaussian (SoG) distributions and perform Cumulative Probability Sampling (CPS) which ensures the entries with higher likelihood to be more likely selected. Graphically, Sampling-Product inference perform sampling inside message constructions. As a result, K probable relative pose estimates can be determined. Therefore, better uncertainty estimates are achieved with a sound probabilistic explanation.

 $<sup>^{2}</sup>$  Note that GPS is unreliable in many places due to occlusions by trees and buildings.



**Fig. 12.** The large-scale SLAM experiments. (a) Robot trajectories of ICP-SLAM and CRF-SLAM before loop closure and uncertainty ellipsis of CRF-SLAM propagated by the filter for every 100 poses. (b) Robot trajectories after loop closure and the uncertainty ellipsis for the relevant poses in (a). (c), (d) The open-loop and close-loop maps of ICP-SLAM overlapped with aerial photo. (e), (f) The open-loop and close-loop maps of CRF-SLAM overlapped with aerial photo.

#### 7. Conclusions

This paper presents an efficient sampling inference procedure to estimate the relative pose uncertainty for laser scan registration with CRF-Matching. In contrast to existing techniques, our approach focuses on capturing the uncertainty from laser point associations. A sampling mechanism is employed during message propagation in a graphical model to produce probable association configuration hypotheses for two consecutive laser scans collected by a moving robot. This enables the computation of uncertainty for both translation and rotation in a sound probabilistic manner. The proposed algorithm is demonstrated in a simulated data set and in a large-scale SLAM for a challenging urban environment.

#### Acknowledgements

The authors wish to thank Tim Bailey, Karl Granstrom and Ian Mahon for numerous helpful discussions.

This work has been supported by the Australian Centre for Field Robotics.

#### References

- F. Lu, Shape registration using optimization for mobile robot navigation, Ph.D. Thesis, University of Toronto, Toronto, 1995.
- [2] F. Lu, E. Milios, Robot pose estimation in unknown environments by matching 2D range scans, Journal of Intelligent and Robotic Systems 18 (1997) 249–275.

- [3] T. Bailey, J. Nieto, Scan-SLAM: recursive mapping and localisation with arbitrary-shaped landmarks, in: Workshop on Quantitative Performance Evaluation of Navigation Solutions for Mobile Robots, Robotics: Science and Systems Conference (RSS), IEEE, Zurich, Switzerland, 2008.
- [4] S.T. Pfister, K.L. Kriechbaum, S.I. Roumeliotis, J.W. Burdick, Weighted range sensor matching algorithms for mobile robot displacement estimation, in: IEEE International Conference on Robotics and Automation (ICRA), vol. 2, IEEE Robotics and Automation Society, Washington, DC, USA, 2002, pp. 1667–1674.
- [5] C.C. Wang, Simultaneous localization, mapping and moving object tracking, Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, 2004.
- [6] O. Bengtsson, A.J. Baerveldt, Robot localization based on scanmatching–estimating the covariance matrix for the idc algorithm, Robotics and Autonomous Systems 44 (2003) 29–40.
- [7] A. Censi, An accurate closed-form estimate of ICP's covariance, in: IEEE International Conference on Robotics and Automation (ICRA), IEEE Robotics and Automation Society, Roma, Italy, 2007, pp. 3167–3172.
- [8] F. Ramos, D. Fox, H. Durrant-Whyte, CRF-matching: conditional random fields for feature-based scan matching, in: Robotics: Science and Systems (RSS), IEEE, Altlanta, US, 2007.
- [9] R. Smith, M. Self, P. Cheeseman, Estimating uncertain spatial relationships in robotics, Autonomous Robot Vehicles 1 (1990) 167–193.
- [10] J. Lafferty, A. McCallum, F. Pereira, Conditional random fields: probabilistic models for segmenting and labeling sequence data, in: International Conference on Machine Learning (ICML), Morgan Kaufmann, Williamstown, US, 2001, pp. 282–289.
- [11] S. Kumar, Models for learning spatial interactions in natural images for context-based classification, Ph.D. Thesis, The Robotics Institute, Carnegie Mellon University, Pittsburgh, 2005.
- [12] R. Cowell, P. Dawid, S. Lauritzen, D. Spiegelhalter, Probabilistic networks and expert systems, in: Statistics for Engineering and Information Science, Springer Verlag, Secaucus, NJ, US, 1999.
- [13] Y. Wang, Q. Ji, A dynamic conditional random field model for object segmentation in image sequences, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, IEEE Computer Society, Washington, DC, USA, 2005, pp. 264–270.
- [14] S. Kumar, M. Hebert, Man-made structure detection in natural images using a causal multiscale random field, in: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, IEEE Computer Society, Madison, Wisconsin, US, 2003, pp. 119–126.

- [15] L. Liao, D. Fox, H. Kautz, Extracting places and activities from gps traces using hierarchical conditional random fields, The International Journal of Robotics Research 26 (2007) 119–134.
- [16] B. Douillard, Vision and laser based classification in urban environments, Ph.D. Thesis, Australian Centre for Field Robotics (ACFR), The University of Sydney, Sydney, 2009.
- [17] J. Friedman, T. Hastie, R. Tibshirani, Additive logistic regression: a statistical view of boosting, The Annals of Statistics 28 (2000).
- [18] Y. Freund, R.E. Schapire, Experiments with a new boosting algorithm, in: International Conference on Machine Learning (ICML), IEEE, Bari, Italy, 1996.
- [19] J. Besag, Statistical analysis of non-lattice data, The Statistician 24 (1975).
- [20] S. Kumar, M. Hebert, Discriminative random fields: a discriminative framework for contextual interaction in classification, in: International Conference on Computer Vision (ICCV), IEEE Computer Society, Nice, France, 2003, pp. 1150–1157.
- [21] S. Friedman, D. Fox, H. Pasula, Voronoi random fields: extracting the topological structure of indoor environments via place labeling, in: International Joint Conference on Artificial Intelligence (IJCAI), Morgan Kaufmann, Hyderabad, India, 2007.
- [22] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann, San Mateo, CA, US, 1988.
- [23] C.M. Bishop, Pattern Recognition and Machine Learning, Springer Verlag, Secaucus, NJ, US, 2006.
- [24] A.P. Dawid, Applications of a general propagation algorithm for probabilistic expert systems, Statistics and Computing 2 (1992) 25–36.
- [25] D. Nilsson, An efficient algorithm for finding the m most probable configurations in probabilistic expert systems, Statistics and Computing 8 (1998) 159–173.
- [26] T. Bailey, Mobile robot localisation and mapping in extensive outdoor environments, Ph.D. Thesis, Australian Centre for Field Robotics (ACFR), The University of Sydney, Sydney, 2002.
- [27] H. Durrant-Whyte, Notes on Monte-Carlo and kernel methods, Technical Report, Australian Centre for Field Robotics, Sydney, 2003.
- [28] R.M. Eustice, Large-area visually augmented navigation for autonomous underwater vehicles, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, 2005.
- [29] J. Callmer, K. Granstrm, Large scale SLAM in an urban environment, Technical Report LITH-ISY-EX-08/4115–SE, Department of Electrical Engineering, Linkping University, 2008.
- [30] F. Lu, E. Milios, Globally consistent range scan alignment for environment mapping, Autonomous Robots 4 (1997) 333–349.