Planning and Localization Using Contacts

Bradley L. Saund

May 2017

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Howie Choset, Co-advisor
Reid Simmons, Co-advisor
Matt Mason
Arun Srivatsan

*Submitted in partial fulfillment of the requirements
for the degree of Masters of Computer Science.*

Keywords:

Abstract

A short summary.

Acknowledgments

My advisor is cool.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

1.2 Contribution

1.3 Thesis Outline

This is some text

Chapter 2

Localization using Contacts

Many robotic tasks require precisely localizing an object, for which tactile sensing is an appealing sensing modality. As motivation, we examine touch localization of a partially manufactured part that requires additional machining operations. In order to handle objects with complex shape, prior information of the object shape is used, and most previous work assumes that the geometry (CAD) model will match the real object exactly.

However, during the manufacturing and assembly processes there are tolerances between different sections of the assembly. A *datum* is defined as a geometric constraint within the object that is used as the reference to define the location of one section of the part with respect to another section. The tolerance is the allowed deviation of the actual manufactured dimensions from the nominal designed dimensions. We assume a part can be divided into precisely manufactured sections, and our method focuses on handling errors due to imprecise machining over large distances and non-critical components, as well as assembly tolerances.

The introduction of tolerance increases the degrees of freedom (DOFs) of the system, as prior to measurement the true dimensions of the full part are unknown. These internal DOFs can be modeled as transformations with uncertainty between sections of the object. For objects with internal tolerances, perfectly localizing a single datum will not necessarily reduce the uncertainty of the full system sufficiently to perform the desired task. On the other hand, it is usually only necessary to localize a subset of the sections of an object.

In this localization problem, the task is to estimate the pose of a goal feature given multiple measurements obtained through probing. These probing measurements are modeled as a Markov process, where each measurement corresponds to a single action/observation pair. This model eliminates the need to store all of the past measurements. We use a particle filter to numerically store and update the belief [**?**].

Figure 2.1 shows a visualization of the initial (2.1a) and final (2.1b) beliefs of the poses of the sections of the object. The task is to drill a hole, shown as a green cylinder, at a specific location defined by offsets from other sections. Internal tolerances prevent simply treating the entire system as a rigid body.

This paper first briefly summarizes the framework of our previous work on rigid-body particle filtering for high-precision localization (section 2.2), which overcomes the particle starvation problem and serves as the basis for this paper [**?**]. We then introduce the datum-based particle filter to generalize this rigid-body approach for objects with coupled rigid sections (section 2.3).

(a) Initial belief

(b) Belief after localization of goal feature (green cylinder)

Figure 2.1: Visualization of the belief of the pose of all sections of the part
(a): The prior belief of the poses before localization. The uncertainty of the goal feature is too high to perform the task.
(b): The belief of the poses after performing measurements to localize the goal feature. The pose of the goal feature is now known well enough to perform the task. Although the bottom edge (purple) and perpendicular section still have noticeable error, precise localization of these features is not needed.

We propose two related but different approaches for this problem. The first approach maintains a single particle filter system that stores the full joint distribution of the coupled datums, while the second approach simplifies the relationships by assuming independence between the distribution of the internal transformations and the pose of the sections, and models the system using separate coupled particle filters.

We also generalize the technique of choosing informative measurement actions to accommodate objects with coupled sections. To achieve this, many potential measurement actions are sampled and the action with the highest expected information gain is selected. Fully predicting the information gain over the continuous belief is computationally expensive, so similar to other approaches [**?**] our work involves a fast approximation for information gain that takes advantage of the discretized belief from the particle filter (section 2.4). We demonstrate our method on a simulated part (section 2.5).

2.1 Current Methods for Localizing Objects Using Contact Sensors

Recently, there have been a variety of approaches that allow robots to localize objects solely with contact sensors. Different contact sensors have been explored and developed, including basic binary sensors, 6-axis force and torque sensors [**?**], soft tactile sensor arrays [**?**], and bio-inspired fingertips [**?**]. Localization with laser sensors has also been used in the high-precision CNC localization, where a 3D point cloud is acquired in order to estimate the transformation between the actual and planned pose [**?**]. Our localization approach can be generalized to these sensors which can distinguish between contact and free-of-contact states.

Particle filters have been widely used and developed since their introduction. Unlike some other Bayesian estimation approaches such as Kalman filters [**?**], extended Kalman filters [**?**] and unscented Kalman filters [**?**], particle filters can easily model non-Gaussian and multi-modal probability distribution. For touch localization, contact sensors yield a highly non-linear measurement model, and the belief can frequently become multi-modal when multiple configurations are all consistent with the measurement. These properties make particle filter a popular approach for touch localization tasks.

However, particle filters will experience *particle starvation* for measurements with very low uncertainty and objects with a high dimensional configuration space [**?**]. Koval introduced the Manifold Particle Filter to address this issue by implementing different sampling and weighting strategies compared to the traditional particle filters [**?**][**?**]. Instead of sampling particles from the process model and weighting them based on the observation, samples are directly drawn from the contact manifold, given the observation. This provides a fast and robust solution for objects with simple shapes. For complex object geometries, as in our case, direct, efficient sampling from the contact manifold becomes difficult.

Petrovskaya tackles particle starvation during touch localization by combined Monte Carlo approaches with annealing as a smoothing technique [**?**]. She introduced Scaling Series algorithms for 6-DOF global tactile localization in both full-constrained and under-constrained scenarios to overcome particle starvation by adjusting particle density depending on the complexity of the posterior. Multiple iterations through the measurement data are used and the precision of the belief is scaled from low to high in order to avoid unnecessarily precise estimates in unlikely regions of belief space.

All of the work mentioned so far assumes the prior information of the object geometry matches the real piece perfectly. Hebert et al. [**?**] solve a touch localization problem using a Bayes filter where an object may have additional unknown parameters describing the shape, such as a screw driver with an unknown length handle. Measurement actions are selected using joint information gain over the object's pose and these internal parameters.

2.2 Rigid-Body Object Localization

The datum-based particle filter described later relies heavily on the framework and methods developed for rigid-body localization developed in [**?**], and this section provides an overview of the relevant details. The task is to determine the pose of an object by choosing and performing touch measurements, given the geometry of the object and prior belief over the distribution of poses.

The geometry of the object to be localized is stored in a STL file using a triangular mesh, defined in the *part frame*. The pose of the object can then be defined as the transformation between this *part frame* and the *world frame* of the workspace. The object is assumed to be fixed in the workspace during measurement and localization, thus the configuration will not change during the localization process.

In order to estimate the true distribution of the pose, each particle in the particle filter represents a single potential pose ${}^{i}x \in SE(3)$ of the object. For a rigid body, the pose includes both translational dimensions (x, y, z) and rotational Euler angles (α, β, γ). We represent the state as a 6D vector $(x, y, z, \alpha, \beta, \gamma)$ in the configuration space. The particle filter updates based on a set of measurements $M_t = \{m_1, ..., m_t\}$ made by the robot directly on the object.

2.2.1 Measurement Model

A measurement action, $\mathcal{M}$ is defined by a start point $\mathcal{A}_p$ for the probe and a linear trajectory vector $\mathcal{A}_v$ both in $\mathbb{R}^3$. The measurement value m is the distance the probe travels in the direction of $\mathcal{A}_v$ until contact is made. The point of contact can then be recovered by $A_p + m\ \frac{\mathcal{A}_v}{||\mathcal{A}_v||}$. The entire information obtained from the measurement t is $z_t = \{\mathcal{M}_t, m_t\}$. Measurement error exists due to sensor error and robot uncertainty.

2.2.2 Modeling Part Uncertainty: Problems with the Standard Particle Filter

At each time step, the belief is a probability distribution of poses, approximated using a particle filter. This belief distribution is also used to plan measurements actions that maximize the expected information gain. The advantage of using a particle filter over other Bayes estimation methods, such as the original[**?**], extended[**?**] or unscented[**?**] Kalman filter, is that a particle filter can model multi-modal non-Gaussian distribution with a non-linear measurement model.

The belief of the state at time t is dependent on the measurements: $bel(x_t) = p(x_t|z_1, \dots, z_t, x_0)$. The measurement model follows a non-linear probabilistic function of the true state: $m_t \sim p(m_t|x_{actual})$. As in all Bayesian filters, the belief $bel(x_{t+1})$ is calculated recursively as follows[1] [**?**]:

$$bel(x_{t+1}) \leftarrow \eta\, p(z_t|x_{t+1})\, bel(x_t) \tag{2.1}$$

with η as a normalization factor. Each new measurement value triggers an update to the belief $bel(x)$.

2.2.3 Rejection Sampling to Overcome Particle Starvation

The traditional particle filter uses importance sampling to update particles [**?**], where samples are drawn based on the process model and weighted by the observation. However, for precise measurements and higher dimensions, the chance that a particle is consistent with the measurement is extremely low, thus few of them will survive during the update [**?**]. Alleviating this problem would require an exponential (in the dimension of the state) number of particles to maintain high enough particle density.

Instead, our approach updates the particles using rejection sampling [**?**]. At each update step t, the continuous prior belief is estimated using Gaussian Mixture Models by applying a Gaussian kernel to each particle, with the kernel covariance proportional to the covariance of the particle configurations. New samples ${}^{i}x_{t+1}$ are then drawn directly from this estimated prior. Using rejection sampling, each sample is accepted with probability $p(z_t|x_{t+1})$. Sampled configurations with the part boundary far from the measured contact location, and thus inconsistent with the measurement, will have a low probability of being accepted

[1] The full update of a Bayesian filter also includes a process model. Our assumption of a fixed object yields the static process model and this simpler formulation

The computational cost of potentially rejecting many samples is offset by using a voxelized distance field [**?**], where the distance between each voxel in 3D workspace and the object is precomputed. As the measurement m_t is described in the world frame, during the rejection sampling the measurement is transformed to the part frame before lookup in the distance field. The minimal unsigned distance $dist_u(m_t, S)$ between each measurement m_t and the object $S(x_{t+1})$ can be obtained directly[**?**]:

$$dist_u(M_t, S(x_{t+1})) = D_f(T(x_{t+1})^{-1}M_t) \tag{2.2}$$

$$dist(M_t,S(x_{t+1})) = \tag{2.3}$$

$$\begin{cases} dist_u(M_t, S(x_{t+1})) - r_p, & \text{if } M_t \notin S \\ -dist_u(M_t, S(x_{t+1})) - r_p, & \text{otherwise.} \end{cases} \tag{2.4}$$

where D_f is the precomputed distance field, r_p is the radius of the spherical tip of the touch probe.

New samples x_t are accepted based on the signed distance $dist(M_t, S(x_t))$ between the object and the measurement on the object. Rejection sampling continues until a desired number of particles have been accepted for this update. These particles represent the updated belief $bel(x_t)$ of the pose.

2.3 Datum Based Particle Filter

The particle filter localization method presented above assumes that the object matches its CAD model exactly. However, this is usually not the case, due to tolerances in the manufacturing and assembly processes. To handle manufacturing deviations, features on parts are not located with respect to the part frame, but with respect to datums, (edges, surfaces, and holes) on the actual "as built" part. Incorporating the notion of datums, and their relationships, adds complexity because the relationship between the datum and the CAD model contains uncertainty. Thus, measuring one section of the assembly provides only uncertain updates to other sections, dependent on the specified tolerances. The following formulation treats these as semi-rigid parts, where each complete part is composed of rigid *sections*, coupled through a probabilistic distribution of transformations connecting the section frames. The datum based particle filter is introduced to allow updates on the belief of all sections of a part using the prior distribution of coupling transformations, and a measurement on a single section.

2.3.1 Datum Representation

The formulation introduced here treats the overall part as composed of separate, known sections. The problem is to precisely localize some feature which cannot be measured *directly* (e.g. a location to drill a hole) with respect to given datums (other sections). To localize the goal feature, certain datums must be localized in certain dimensions. For instance, Figure 2.2a shows a hole feature referenced to the top and right edges datums of our part. The true part configuration is shown in gray in 2.2b and 2.2c. In this example, it is necessary to localize the top edge's vertical position and orientation, but not its in-page or horizontal position. Similarly, the right edge only must be localized horizontally.

We introduce two approaches; the first explicitly represents the joint probability distribution between the sections, and the second stores separate, independent probability distributions for each section. Figures 2.1 and 2.2 visualize the independent-state particle filter. The full-state particle filter produces similar images. We continue to assume that the measurements will be made by point-based sensors.

Throughout the rest of this paper we will use the following notation. X_t^k is the set of N particles representing the belief of section k at time step t. Frequently t is omitted when implicit. Each particle is a configuration for a single section $X^k = \{^j x^k\}_{j=1}^N$. The omission of k indicates all necessary particles to represent the belief of the part: $X = \{X^k\}$ and $x = \{x^k\}$.

2.3.2 Geometric Relationships

Geometric relationships are defined between two or more part sections. The existence of the tolerance introduces uncertainty to these relationships, which are modeled as a distribution of transformations in the configuration space between the pose of each section. More generally, the conditional probability $p(x_t^k|\mathbf{x_t})$ represents the belief of the pose of section k given the poses of the other sections.

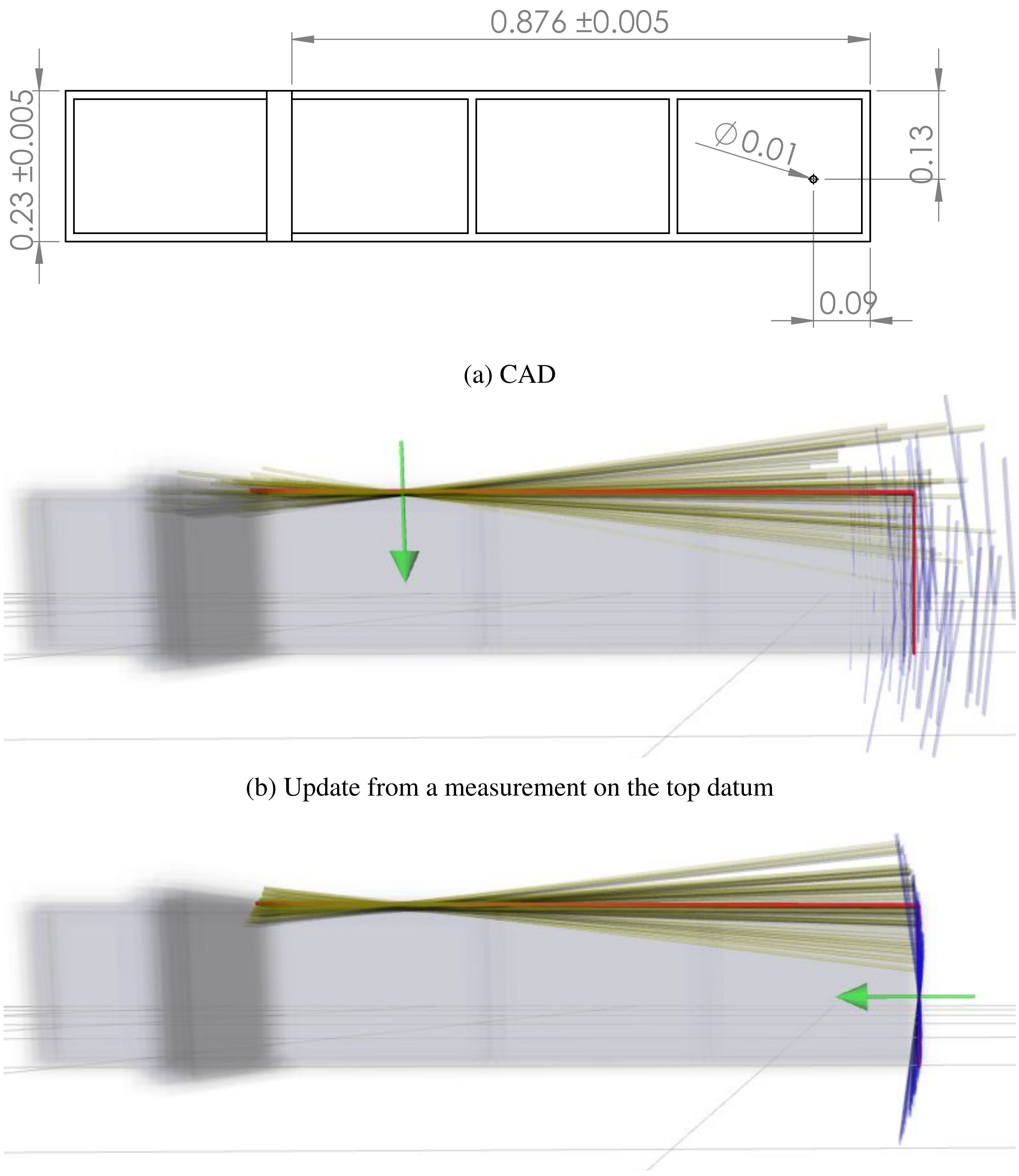
A measurement is made on a single section at each step by a touch probe. Let the measurement on the section k at step t be $\mathcal{M}_t^k$, then the posterior of section k is $p(x_t^k|\mathbf{x_t}, M_t^k)$. In the following algorithms, the section that a measurement contacts is known. This assumption is reasonable if the uncertainty of the prior belief is small compared to the physical size of each section, and measurement are not chosen on the boundary between sections. If this assumption does not hold, localization can be performed for the whole object using the methods of section 2.2 to get better estimate, before considering the object as a combination of coupled sections.

2.3.3 Full-State Representation

The first method we present maintains the full joint probability distribution between sections using a single set of full-state particles: $X_t = \{^j x_t\}$. Thus, each particle represents a combination of the poses of all of the sections (see Fig. 2.3), which is drawn from the prior joint distribution $bel(x_0)$.

Instead of applying a Gaussian kernel in 6D configuration space as done in section 2.2, the continuous prior is estimated by Gaussian Mixtures in the full $6n$-dimensional configuration space, with the kernel covariance proportional to the covariance of the sampled states. This is achieved by applying kernel density estimation techniques. One popular method for the bandwidth selection is the Silverman's rule-of-thumb estimator[**?**]. Other techniques are discussed in more detail in [**?**].

Although it seems at first that the high dimensional state space will require a prohibitively huge number of particles to approximate the true distribution, in practice, we have not found this to be the case. Largely, this is due to the small internal tolerances compared to the uncertainty of the pose of the object as a whole, so particles tend to cluster in a small subset of the full state space. Note that if there were no uncertainty in the transformation between sections, then the



(a) CAD

(b) Update from a measurement on the top datum

(c) Update from another measurement on the right datum

Figure 2.2: Visualization of independent-state particle filter
(a): Side view of the CAD drawing with dimensions (simplified for clarity). This drawing indicates the nominal distance between the top and bottom edge is 0.23m, with a symmetric tolerance of 5mm. This drawing also defines a hole with a 1cm diameter, and the top edge as its vertical datum and side edge as its horizontal datum.
(b): The beliefs of the top (green) and right (blue) edges of the part are shown. The true part location is shown in gray. The measurement (arrow) on the top section partially localized the top edge. For clarity in the image, the belief of the other sections are not shown, and only 50 of the 500 particles are shown.
(c): A following measurement (arrow) on the right edge further localizes the part. This measurement provides information on the right edge directly, and the top edge indirectly.

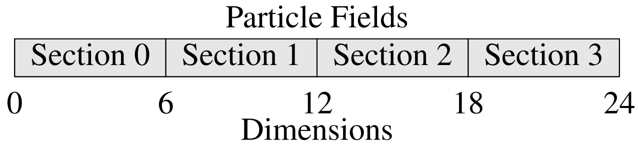


Figure 2.3: A 24-dimensional particle for object with 4 sections

Algorithm 1 Full-State Particle Filter

```
Input: number of particles $N$ and number of sections $n$
Input: particles $\mathbf{X_t}$
Input: observation $m_t^i$
Input: meshes $S = \{S_k\}_{k=1}^n$
Output: particles $\mathbf{X_{t+1}} = \{\mathbf{^j x_{t+1}}\}_{j=1}^N$
build distance field $D_f(p)$
$j \leftarrow \mathbf{1}$
while $j \leq N$ do
    $\mathbf{x} \sim p(x|\mathbf{X_t})$                                  $\triangleright\ x = \{x^k\}_{k=1}^n$
    $dist \leftarrow D_f(T(x^i)^{-1} m_t^i)$
    if $dist \leq \xi$ then
        $^j x_{t+1} \leftarrow x$
        $j \leftarrow j + 1$
    end if
end while
```

system reduces to a single 6 dimensional state space. In addition, the Gaussian kernel applied to the particles is capable of creating broad beliefs in specific dimensions[**?**], thus this particle filter is able to model the mixture of precisely localized, and poorly localized dimensions. Finally, the sample acceptance probability, $p(z_t|x_t)$, depends only on the 6 dimensional subspace corresponding to the pose of the section being touched, and although only a thin manifold is accepted with high probability, the particle starvation challenge is no worse than previously addressed in 2.2.

At any time step t the belief of the state $bel(\mathbf{x_t}) = p(\mathbf{x_t}|Z_t, x_0)$, and x_t^k is the portion of x_t representing the pose of section k. During rejection sampling, new full-state samples x_{t+1} are drawn from the estimated $\hat{bel}(\mathbf{x_t})$ directly. The probability of accepting a sample is based on the measurement M_t^k on section k, and is computed by extracting the 6D pose x_{t+1}^k from each full-state sample x_{t+1}. Equation 2.2 and 2.3 are then used for the rejection sampling, except that the transform from the extracted x_{t+1}^k is used to transform the measurement. If $dist(M_t^k, S_k(x_{t+1}^k))$ is sufficiently large, this full-state sample $\mathbf{x_{t+1}}$ is rejected, otherwise accepted. Note that the full-state sample $\mathbf{x_{t+1}}$ is accepted based on $p(M_t^k|x_{t+1}^k)$, and the CAD model used for distance field k is the mesh for that particular section, S_k (shown in Algorithm 1). For simplicity, the pseudo-code does not include the adaptively adjusted sample size based on KL-divergence[**?**][**?**].

Algorithm 2 Independent-State Particle Filter

```
Input: number of particles $N$ and number of sections $n$
Input: sets of particles $\mathbf{X_t} = \{X_t^k\}_{k=1}^n$
Input: observation $m_t^i$
Input: meshes $S = \{S_k\}_{k=1}^n$,
Input: transformations $\{p(T_i^k)\}_{k=1}^n$
Output: particles $\mathbf{X_{t+1}} = \{X_{t+1}^k\}_{k=1}^n = \{{}^j x_{t+1}^k\}$
build distance field $D_f(p)$ for section $S_i$
for $k = 1, ..., n$ do
    $j \leftarrow 1$
    while $j \leq N$ do
        $x \sim p(x^k | X_t^k)$
        $T_i^k \sim p(T_i^k)$
        $\tilde{x} \leftarrow T_i^k \times x$
        $dist \leftarrow D_f(T(\tilde{x})^{-1} M_t^i)$
        if $dist \leq \xi$ then
            ${}^j x_t^k \leftarrow x$
            $j \leftarrow j + 1$
        end if
    end while
end for
```

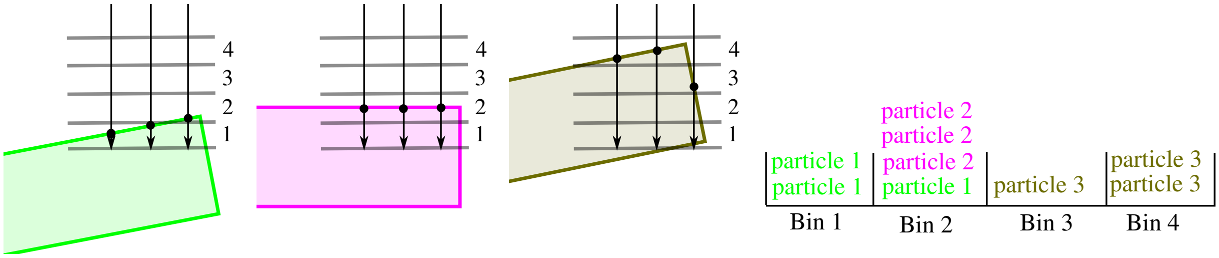
2.3.4 Independent-State Representation

The first approach described above tracks the updates of both the pose of each datum and the their transformations by maintaining a full-state representation of the distribution. An alternative is to maintain the probability distribution for each section separately. Instead of using a full high-dimensional particle filter for the full object, individual 6-dimensional particle filters are used for each individual section under the approximation that the belief over transformations between sections are fixed and independent. While this loses information compared with the full joint belief, in practice this loss is acceptable.

As in the rigid body particle filter described in 2.2, a sample in the particle filter for section k represents a $SE(3)$ pose of the geometry of section k. The transformation information between different sections are defined explicitly. The prior belief on the transformation from section k to section j is $bel(T_k^j)$, which is a distribution over $SE(3)$ transformations. A measurement on a single section updates all individual particle filters related through a defined transformation distribution. Given a measurement M^k on the section k, the updated belief becomes $p(x_{t+1}^j | T_k^j, M^k)$ for a related section j.

The update to the belief $bel(x_t^k)$ given a measurement performed on section k itself is identical to the particle filter update for the rigid object. Since T_k^k is the identity with probability 1, the updated belief can be written as:

$$bel(x^k) = p(x^k | T_k^k, M^k) = p(x^k | M^k) \tag{2.5}$$



(a) Part in configuration: particle 1 (b) Part in configuration: particle 2 (c) Part in configuration: particle 3 (d) Binned Measurements

Figure 2.4: Binning of potential measurement on particles for use in calculating information gain. The three arrows represent the nominal measurement action $\mathcal{M}$ with simulated deviation δ_j. The horizontal lines divide the measurement values into the numbered bins.

For a section j that references section k ($k \neq j$), each new sample x^j_{t+1} is drawn from the prior of its corresponding particle filter j. x^j_{t+1} is then transformed from the frame of section j to the frame of section k:

$$\tilde{x}^k = {}^iT^k_j \times x^j \tag{2.6}$$

where ${}^iT^k_j \sim bel(T^k_j)$ is a sampled transformation from the distribution $bel(T^k_j)$. As the measurement was performed on section k, the geometry of section k is used rather than j when computing the consistency with the measurement. The sampled particle is accepted with probability $p(M_t|\tilde{x}^k)$. The above process is repeated until the desired number of particles have been accepted (shown in Algorithm 2).

2.4 Predicting Effective Measurement Actions

Performing measurements is expensive, so we choose the measurement action that provides the most *information gain* on the goal feature. We treat each action as a probabilistic decision over a set of particles approximating the belief of the goal feature. This is an approximation for the information gain for the underlying, continuous belief distribution. The best measurement may not be on the goal feature, and it may be impossible to even measure the goal feature directly. Our formulation predicts the information gain on the goal feature for both a measurement directly on the goal feature, or indirectly for a measurement on datums or other sections of the part.

2.4.1 Information Gain

Given X^G, a set of particles representing the belief of the goal feature, the *information gain* from a measurement action $\mathcal{M}$ is defined as the expected reduction of entropy.

$$IG(X^G|\mathcal{M}) = H(X^G) - H(X^G|\mathcal{M}) \tag{2.7}$$

$H(X^G)$ is the entropy of the particles and $H(X^G|\mathcal{M})$ is the entropy of the particles conditioned on the measurement action.

The entropy of a discrete distribution of states depends only on the probabilities of each state occurring.

$$H(X^G) = -\sum_i w_i \log w_i \tag{2.8}$$

where w_i is the weight of particle i.

To calculate the conditional entropy, $H(X^G|\mathcal{M})$, the measurement action $\mathcal{M}$ is simulated on the part distribution. Performing a measurement action yields a continuous distribution of measurement value. $\mathcal{W}$ samples are drawn from this distribution for each particle:

$$m_{i,j} = Simulate(\{\mathcal{M} + \delta_j\}, {}^i X^G) + \eta_j \tag{2.9}$$

$$j \in \{1, 2, \ldots, \mathcal{W}\} \tag{2.10}$$

$$i \in \{1, 2, \ldots, N\} \tag{2.11}$$

where δ_j is the deviation from the nominal measurement action, $Simulate$ computes the value for a measurement action applied to the part in a specific configuration, and with η_j as measurement noise.

$H(X^G|\mathcal{M})$ is calculated by dividing the continuous values $m_{i,j}$ into discrete bins, b_k. The conditional entropy of this measurement action is then:

$$H(X^G|\mathcal{M}) = \sum_k p(b_k)\ H(X^G|b_k) \tag{2.12}$$

where $p(b_k)$ is the prior probability that this measurement will fall into bin b_k and $H(X|b_k)$ is the entropy of the particles within bin b_k. The likelihood of a bin is computed by summing the weights of the measurements in that bin. Defining the weight of the bin, W_k as:

$$W_k = \sum_{i,j} \mathbb{1}(m_{i,j} \in b_k) \cdot w_i \tag{2.13}$$

then:

$$p(b_k) = \frac{W_k}{\sum_{i,j} w_i} \tag{2.14}$$

$$= \frac{W_k}{\mathcal{W}} \tag{2.15}$$

Given a bin, the probability of a specific particle is:

$$p({}^i X|b_k) = \frac{\sum_j \mathbb{1}(m_{i,j} \in b_k) \cdot w_i}{W_k} \tag{2.16}$$

Then the entropy of the bin can be calculated:

$$H(X|b_k) = -\sum_i p({}^i X|b_k) \log(p({}^i X|b_k)) \tag{2.17}$$

Figure 2.4 visualizes this binning process. Three measurement actions (arrows) $\mathcal{M} + \delta_j$ are simulated on three configurations of the part (${}^i x$ $=$ particle i). The intersection between the simulated measurement action and the part determines the measurement value $m_{i,j}$. These measurement values are sorted into bins.

Adaptations are made to this process to accommodate our two representations as described next:

2.4.2 Information for Full-State Representation

The full-state representation does not maintain a set of particles over just the goal feature, but rather each particle, X, represents the full $6 \times n$ state. Using these full-state particles for X^G above provides a good metric for localizing every section of the part, but a poor metric for localizing the goal feature. This metric would often suggest to perform measurements on non-datum features that are irrelevant to the location of the goal feature. The error in this metric is due to

$$H(X^G|\mathcal{M}) \neq H(X|\mathcal{M}) \tag{2.18}$$

One approach is to incorporate domain knowledge when designing the full-state representation, by including only the relevant datums necessary for a particular task. Then, any information on this limited full state will be reduction of uncertainty of at least one datum required for the task.

An approach that does not require pruning irrelevant sections from the full state involves combining particles that produce similar configuration for the goal feature. To calculate $H(X^G|\mathcal{M})$ using particles X, measurement actions are used to sort the particles into bins as in described in section 2.4.1. Then, particles that produce sufficiently similar goal feature configurations are treated as identical particles when computing entropy, by combining these into groups L.

$$p({}^L X|b_k) = \frac{\sum_{i,j} \mathbb{1}(m_{i,j} \in b_k) \mathbb{1}(i \in L) \cdot w_i}{W_k} \tag{2.19}$$

$$H(X|b_k) = -\sum_L p({}^L X|b_k) \log(p({}^L X|b_k)) \tag{2.20}$$

These group can be constructed by discretizing the space of possible configurations for the goal feature. An issue which this paper does not address is the balance of a requiring a reasonably small number of particles while maintaining sufficient density for this descretization of goal feature configuration to produce meaningful group sizes.

2.4.3 Information for Independent-State Representation

The independent-state representation does maintain the set of goal feature particles, X^G, however additional steps are needed when computing Eq. 2.9. When simulating $\mathcal{M}$, the robot will measure some section, $\mathcal{S}$, of the part. Simulating the measurement using $X^{\mathcal{S}}$ is straightforward, but leads to computing $IG(X^{\mathcal{S}}|\mathcal{M})$, which is not the desired metric $IG(X^G|\mathcal{M})$.

The independent-state representation makes the approximation that the distribution of transformations between sections are fixed and independent, and this approximation is used to achieve the desired metric. A temporary set of particles $\tilde{X}^{\mathcal{S}}$ is created by sampling transforms $T_G^{\mathcal{S}}$ and applying these transforms to X^G. $\tilde{X^{\mathcal{S}}}$ is used in Eq. 2.9 to generate sample measurement values $\tilde{m}_{i,j}$, which are used in the calculation for bin entropy $H(X|b_k)$. While the independence approximation could be used again in the calculation of bin probabilities $p(b_k)$, this approximation is not needed. Measurements $m_{i,j}$ calculated using $X^{\mathcal{S}}$ are used to calculated $p(b_k)$.

2.5 Experiments

We validated both the full-state representation approach and independent-state representation approach in simulation. The software was implemented in ROS using C++. In our experiments, we simulated a specific task which is common in manufacturing: localizing a target location, defined by datums, to drill a hole on an object. The datum-based particle filter was simulated on a structural component used in aircraft. This is the same object as used in the original rigid-body particle filter paper[**?**], with adaptations made to allow internal degrees of freedom. The object is composed of 5 precisely manufactured sections, and tolerance between sections was determined by engineering drawings (for precisely defined features), and our guesses (for loosely defined relationships).

2.5.1 Measurement Selection

The target hole is localized by measuring its referenced datums. Specifically, the pose of the hole feature (green cylinder) in figure 2.1 is defined by an offset distance from the top and right sections shown in figure 2.2a, and the axis of the hole is orthogonal to the front plane. The hole does not exist yet, and thus cannot be measured directly. In order to localize the hole location precisely without direct measurement on the hole, we assume that the transformations between the target location and its defining datum sections have very small uncertainty along some dimensions, e.g. the vertical distance between the center of the hole and the top plane.

At each step, the measurement is simulated using ray-mesh intersection algorithms[**?**]. Potential measurement actions are sampled in the workspace. The information gain for each measurement is calculated based on the current estimated pose of each section. In our experiments, for each measurement performed we evaluated candidate actions until we had modeled 500 actions with non-zero information gain. Only the measurement action with the largest expected information gain is "performed" in simulation and used to update the belief.

2.5.2 Simulation Results

We evaluated our two proposed approaches with similar settings. A total of 20 measurements are simulated during each trial. For the full-state representation, a maximum of 800 particles are used. For the independent-state representation, a maximum of 500 particles are used for each section. The simulation uses 5 mesh models for different sections. After each update, the average estimated pose of the hole is computed by averaging the hole poses produced from all particles.

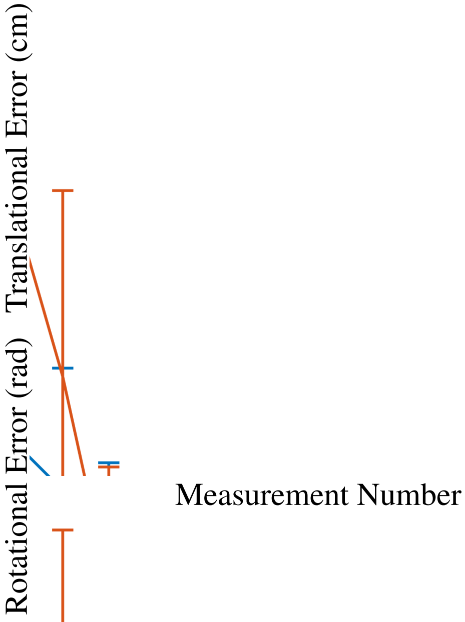


Figure 2.5: Comparison of the accuracy of the update step when using full-state particle filter and independent-state particle filter

Figure 2.5 shows the comparison between the full-state particle filter and independent-state particle filter. Translational error and rotational error are defined between the estimated pose of the hole and its true state. From the simulation, the errors of the estimated pose decreases rapidly for both approaches after each new measurement is applied on the system.

Chapter 3

Planning with Contacts for Support

3.1 Discrete Selection of Contacts

3.1.1 Walking

3.1.2 Grasping

3.2 Trajectory Optimization

3.2.1 Methods for Finding the Minimum Cost Trajectory

3.2.2 Smoothing the Cost Function to Discover Contacts

Auxillary Variables

3.2.3 Calculating Contact Forces

3.3 Sample Based Planning

3.3.1 RRTs and their variants

3.3.2 Adaptation to Encourage Contacts

3.4 Experiments

3.4.1 Simulation and Robot Model

3.4.2 Robot Snakes on a Plane

Chapter 4

Conclusion and Future Work

This paper introduced the *datum-based particle filter*, which provides a method to localize a task location defined by datums on an object with internal degrees of freedom. This method stores the belief as the belief over the poses of multiple rigid sections comprising the object using a particle filter, and selects measurement actions using the metric of information gain. Two implementations are described: a high dimensional particle filter capturing the full state, and multiple particle filters coupled through the tolerances between sections. The techniques to avoid particle starvation during rigid body localization have been extended to both implementations of the datum-based particle filter. Information gain of a potential measurement action is approximated as a discrete probabilistic decision process over the particles comprising the belief. The formulation presented distinguishes between useful information which updates the belief of the target feature, and information which only improves the belief of non-datum sections.