2013 IEEE International Conference on Robotics and Automation (ICRA)
Karlsruhe, Germany, May 6-10, 2013

Uncertainty Aware Grasping And Tactile Exploration

Stanimir Dragiev[κ,φ] Marc Toussaint[φ] Michael Gienger[χ]

***Abstract*— The perception of the surrounding world depends on noisy sensors which introduce uncertainty. When we develop algorithms for grasping with robotic hands it is not enough to assume the best estimate of the environment – if there is a measure of uncertainty we need to account for it. This paper presents a control law which augments a grasp controller with the ability to prefer known or unseen regions of an object; this leads to the introduction of two motion primitives: an explorative and exploitative grasp. We integrate this control law in a framework for iterative grasping and implement a tactile exploration scenario. The experimental results confirm that using the notion of uncertainty in the control loop yields better models and does it faster than an uninformed controller.**

I. Introduction

Grasping previously unknown objects is still a challenge for robotic systems. Consider the situation where a human is reaching into a bag (or is deprived of its visual modality), fumbling for a previously unknown object. The first tactile feedback will allow for a rough estimation of where the object is, leaving large uncertainty about its full extension and shape. With iterated regrasps the human will be able to very quickly reduce these uncertainties and develop a rather precise estimate of its shape and perhaps identify the object from the haptic data only. In this paper we essentially want to develop methods for a robot to perform a similar task, grasping previously unknown objects by relying only on haptic data using an iterative haptic exploration and grasping strategy. This involves addressing several issues: 1) how to represent the current estimate and uncertainty (belief) about the object, 2) how to design active learning strategies for the robot to decide on regrasps, and 3) how to integrate these in a closed sensory motor control loop.

The insight that the object model needs to benefit from the information gained from unsuccessful grasps suggests an iterative scheme for reducing uncertainty. For this to work, in the first place, we need to organize the knowledge about objects in a way which offers this information, i.e. we need a representation which inherently cares about uncertainty. Introducing the notion of uncertainty assigns levels of interest to particular surface regions. Then, we can define control laws which either minimize uncertainty and *grasp at known regions*, or go to the most uncertain regions and thus *more effectively collect information*:

1) Grasp at known regions: Consider a naïve (not aware of model uncertainty) grasp controller which takes as input a model of an object and finds a valid grasp. Assume the controller is correct in the sense that the more precise the model it operates on, the better the performance. Then, the success rate of the controller depends on how precise the input is. Note that not all points are equally important for a valid grasp: apart from collision avoidance, the contact points are the ones which likely render the grasp valid or not. Then, if we minimize the variance at the fingertip contact points, while keeping the grasp feasibility constraints of the controller, we are more likely to successfully grasp, because of the correctness assumption. Thus, if a quantification of uncertainty is available we can define control laws which aim at more robust grasping.

2) More effective and grasp oriented exploration: Variance awareness in the control can serve to collect information about unseen regions of the surface: maximising the variance would lead to a less likely successful grasp, but would bring the benefit, that in case of failure the model would be updated with more information gain, reducing the uncertainty by a larger portion. In addition, exploring objects in this manner collects information which is relevant for the grasp controller, for instance, the fingers of a hand would approach from opposite directions and explore opposing regions. In other words, a grasp controller can *maximize the information gain with respect to its own constraints*, rather than sampling the world randomly or only exploring a neighbourhood at a time. This is yet another manifestation of the insight that the notion of embodiment of the learning system provides the right restrictions to the search space to make learning feasible.

A robot equipped with an uncertainty aware controller should perform manipulation tasks more robustly; when exploring its environment it will gain quality of estimates and convergence speed – and this will increase its overall performance.

The scenario which we pursue is motivated by tactile exploration when other sensor modalities are not available. For instance, structured light sensors, camera, laser are often cheaper and/or more precise than tactile sensors and can be applied passively and from longer distance. But they rely on the presence, diffusion and reflection of light. Still, there are environments where the light conditions are bad due to a weak light source, inappropriate reflections, or bad spread conditions for light, e.g. in the presence of smoke. Another problem for light based sensors is occlusion – when an obstacle blocks the light reflected from object region on the way to the sensor. In these cases tactile feedback from the actuators can complement the other perception modalities.

In the following we first review previous work related to grasping under uncertainty and in III describe briefly our representation of choice. Section IV presents the control laws and we show an implementation of the paradigm which we

[φ]Stanimir and Marc are with the Machine Learning and Robotics group at the FU Berlin, Germany: ssd, mtoussai @mi.fu-berlin.de
[χ]Michael is with Honda Research Institute Europe, Offenbach, Germany. michael.gienger@honda-ri.de
[κ]Stanimir gratefully acknowledges the financial support from Honda Research Institute Europe as a member of CoR-Lab, Bielefeld.

use to obtain the experimental results discussed in V.

II. Related work

A. Representations for object estimation and grasping

A large variety of object representations in the context of grasping have been developed with different foci in mind. To name several examples, polygon meshes are suitable for visualisation and collision test; superquadrics used as primitives for shape approximation [14] feature very compact knowledge representation but with less precision for complex objects; unions of superquadrics or other building blocks as well as voxel and octree representations [8] offer varying precision of approximation by discretising the euclidean space; Direct visual cues [10] are an example for representation directly used for grasp synthesis.

However, object manipulation involves two directions of information processing: sensory feedback and motor control. The above mentioned shape approximations and cues do not fully account for this. A good representation has to 1) provide means for incorporating all available information: prior shape knowledge and heterogeneous uncertain sensor feedback from different modalities need to be fused in a coherent way; 2) the resulting shape belief needs to be in a form convenient for the synthesis of a reach-and-grasp motion. These requirements on representation are motivated also by research about the human sensory and motor system from the fields of psychology and neurobiology. Intraub [5], for instance, suggest that pure unimodal sensory representations alone cannot explain boundary extension in human scene perception.

Gaussian process implicit shape potential (GPISP [3]) comes closest to the requirements we pose. The representation is based on implicit surfaces [2], a representation traditionally used in computer graphics and preferred for its smooth nature and the ability to deal analytically with object deformations. Controllers based on GPISP interpret the implicit surface as potential field and see the effectors as particles under the impact of the field which allows to position and orient them relative to the object.

Commonly, the information about objects comes at different levels of uncertainty. Sensor fusion requires to have means to deal with sensor uncertainty in the representation. Machine learning provides a variety of ways to handle noisy input, some of them used also in shape approximation algorithms, including the Support Vector Machine (SVM) approach by Steinke et al. [12], and Williams and Fitzgibbon [13] who employ a Gaussian Process (GP) in a similar setting to [12]. GPs [9] offer a regression method which takes inputs, potentially with different noise distributions, and produces estimates with error bars. The above works prove the suitability of established ML tools for object estimation – a sane estimate is necessary for the successful interaction with the environment – but deal with uncertainty merely in passive manner. The controller proposed in [3] works on the mean of the shape distribution, i.e. on the maximum a posteriori (MAP) estimate of the shape. It practically ignores one of the gains of the GP – the error bars – which are the focus of the present work.

B. Reducing uncertainty

Having a notion of model uncertainty allows to more effectively search to improve the model. A straightforward approach to reduce the uncertainty is to not rely on a single grasping attempt but rather take into account multiple tries. Every encountered disagreement between the real object and the model can be used to improve the estimate. Following this schema Hsiao et al. [4] take a POMDP approach for inferring by repeated execution of grasping primitives an uncertain object pose. On every execution they learn from unsuccessful grasps – due to collision or miss – and update the belief for the next execution. The uncertainty is limited, though, to the 2D-position of a *known* object relative to the robot.

Not only in grasping context we can search for particular interactions which are more advantageous for specific goals. For instance, for discovering the kinematic structure of articulated objects Katz and Brock [6] evaluate effects of the robot's own movements and find the types of joints that connect rigid bodies.

C. Reactive grasping

The effectiveness of human use of tactile feedback for grasping suggests that in biological entities the sensory and motor pathways are tightly connected. It provides inspiration to seek to integrate them in artificial systems for which we expect to posses advanced motor skills, too. Various ways to do this include integration of tactile feedback in the control loop, visual servoing, grasp success assessment based on tactile events. For instance, the integration of tactile feedback and grasping experience is discussed by Steffen et al. [11], too. Their policy reacts to tactile events during grasping and selects from the experience database the most appropriate target posture according to similarity function based on the current contact configuration.

D. Active learning

In active learning, the system evaluates current evidence to select a point in the space they learn in as a next query (attempt). Using data in form of demonstrated or experienced grasps allows to select the next point in the space of grasps. Kroemer et al. [7] define hierarchical controller – a lower level for executing grasps, and higher for determining the next grasp to be executed. Their decision policy is to maximize the sum of 2 terms: the expected immediate reward (value) of a grasp; and the second is the standard deviation at that point. This policy ensures convergence to a grasp only when no other grasp is more promising (under the assumption that finding global maxima is feasible).

E. Exploration for grasping

Object exploration for acquiring grasp hypotheses has been investigated by Bierbaum et al. in [1]. The authors take an approach which share some keywords with our: they fill the volume of the cube containing the object with attracting points which gives rise to a potential field. They employ the same idea of particles attached to the fingers and describe a policy for creating and deleting potential sources and for changing attracting to repulsive potentials depending on registered contacts, thus leading the end-effector to unexplored areas. The contact points are used to build a polygon mesh and some heuristics are used to evaluate candidate grasps for that kind of representation. Summarized, our heuristics for feasible grasping are different, because they operate on different representation: we use for grasping the very same representation we use for sensing and for motion generation. Furthermore, the resulting potential field is different, since we view it as generated by the whole object; in a sense, the field would match if we infinitely densely covered the surface of the object with attractors and removed the ones outside the object and the repelling ones.

III. Background: GPISP for shape estimation and grasping

GPISP is intended to be suitable for two domains which are often considered separately: object perception and grasp generation. For the sake of completeness of the present paper, we briefly explain how it processes sensor data and furthermore give an intuition of how the representation can be used for grasp control.

GPISP is a composition of the following concepts: It uses *Gaussian process* (GP), a machine learning formalism for non-linear function estimation, to estimate from sensor data a function which implicitly defines the surface of an object as its zero-levelset, an *implicit surface* (IS). The estimated function is so designed that it can be interpreted as a *potential field* generated by the object – monotonously increasing with the distance to the surface. With this interpretation, the value and direction of the potential field can be used to navigate an endefector towards the surface and specify its relative orientation to the object.

A. From observations to shape

The input for a GPISP are points from the surface of an object and the normals to the surface at that points. Before seeing any evidence, the GP is initially biased to some positive value to express the lack of knowledge about objects in the space, $\mu(\mathbf{x}) = 1$. Data points condition the GP estimate to go through the zero line at the observation points: $f(\mathbf{x_s}) = 0$. In addition, the surface normal at the observation point can be incorporated in the GP as the gradient of the function, $\nabla f(\mathbf{x}) = \mathbf{n_s}$. This information can be obtained from, e.g. visual or haptic sensors. A GP potentially allows to even mix modalities with different precision. Note that structured light sensors (depth sensors), stereo vision and other sensing modalities which provide point clouds can be used too, using a point observation directly, and estimating the corresponding normal from neighbouring points and camera position. Adding observations incrementally improves the GP estimate for the function, and its zero-levelset resembles more closely the original object surface.

B. Potential field grasping

The resulting estimate is a Gaussian probability distribution over functions, with particular mean function being the MAP estimate. Interpreting the mean as potential field we can define task space features which describe feasibility heuristics for grasping, e.g.

- move wrist towards the object,
- orient wrist with the inner side to the object
- bring fingertips to the surface
- orient each finger to align with the surface locally, thus maximizing the contact area

These features, weighted according to their importance, define a criterion for a feasible grasp. The optimum of the so designed function provides a good guess for how to grasp an object.

IV. Uncertainty aware grasping

We perceive the world through sensors. Sensors are noisy, i.e. we are uncertain about their readings. Yet, we use these readings to build models of the world and to interact with it. Thus, these models and our actions are uncertain. But to what degree? If we knew the answer, can this help to improve our actions?

A. From sensor uncertainty to model uncertainty

A coherent way to translate sensor uncertainty to model uncertainty is to transform the sensor noise distribution into noise distribution in the space of models. This, precisely, is what GPISP does: the models are Gaussian distributions over functions, the mean being the MAP estimate: $f \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$.

GPISP approximates a function $f(\mathbf{x})$ whose 0-levelset is the surface of the approximated object, $S = \{\mathbf{x} : f(\mathbf{x}) = 0\}$. Together with an estimate $\bar{f}(\mathbf{x})$ of $f(\mathbf{x})$ at arbitrary point $\mathbf{x}$, the GP provides a quantification of the uncertainty about this estimate – the variance of the estimate, $\mathbb{V}[\bar{f}(\mathbf{x})]$. The variance is low near observation points and increases in unseen regions. It is a function of the parameters of the GP covariance. They would be set typically prior to estimation dependent on the precision of the used sensor: the more precise we believe the data is, the less variance at observation points and overall.

Following roughly the notation of Rasmussen and Williams, the variance is given by:

$$\mathbb{V}[\bar{f}(\mathbf{x}_*)] = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}^T \mathbf{G}^{-1} \mathbf{k}. \quad (1)$$

with query point $\mathbf{x}_*$, estimate $\bar{f}(.)$, GP covariance function $k(.,.)$, vector $\mathbf{k}$ of covariances between $f(\mathbf{x}_*)$ and the observations, Gram matrix $\mathbf{G}$ containing the covariances between the observations.

For the GPISP we use Gaussian covariance $k(\mathbf{a}, \mathbf{b}) = \sigma_p^2 \exp\left(-\frac{1}{2\sigma_w^2}(\mathbf{a}-\mathbf{b})^T(\mathbf{a}-\mathbf{b})\right)$. In the presence of noise (uncertainty) the Gramian is indeed augmented with the diagonal matrix $\mathbf{\Sigma}_n$, which has on its diagonal the variance of the Gaussian noise assumed for the particular observation: $\mathbf{G} = \mathbf{K} + \mathbf{\Sigma}_n$.

Note that the entries $\mathbf{k}_i$ of the covariance vector can be expressed in terms of $k(.,.)$ following the result from [9, p.191]. With this we can analytically compute the variance in the case of Gaussian covariance.

B. Uncertainty aware grasp control

Now that we can measure how certain the model is in particular region, we can use this to improve the quality of grasps or grasp sequences.

1) Model variance as motion feature: Introducing variance to control enables us to specify a range of controllers which consistently define exploration/exploitation levels for grasping: the resulting uncertainty aware controller implicitly contains the control costs and they can be traded off for satisfying the variance maximisation (or minimisation).

In the context of GPISP, it is natural to realize such control laws by following the gradient of the variance. For a GP with Gaussian covariance we derive it analytically and can use it as a motion feature over the finger tips' euclidean positions. For instance, consider the motion feature $\mathbf{y} \in \mathbb{R}^d$, the vector of GP variances at d body points. We will call a differentiable motion feature equipped with Jacobian and target value a *task variables* (TV); we introduce TVs in Section IV-D.2 and use combinations of them to define more complex goals.

Before we proceed, we clarify some terms and notation: $\mathbf{q}$ denotes the vector of joint angles (point in configuration space); for each finger we have its current position computed from forward kinematics $\mathbf{x_i} = \phi_i(\mathbf{q})$; we also assume its Jacobian is known: $\frac{\partial \phi_i(\mathbf{q})}{\partial \mathbf{q}} =: \mathbf{J_i}(\mathbf{q})$; $\psi_i = \psi(\mathbf{x_i})$ is the potential field read out, and the variance at that point is $\mathbb{V}_i$. Later, $\mathbf{k}$ is again the vector of covariances between particular point (here ϕ_i) and all GP observations (and $\dot{\mathbf{k}}$ the vector of their derivatives).

The value, $\mathbf{y}$, at each time step is simply the vector of variances $\mathbf{y} = \mathbb{V}_{1:d}$. The Jacobian of a component of the variable $\mathbf{y}$ is:

$$\frac{\partial \mathbf{y}_i}{\partial \mathbf{q}} = \frac{\partial \mathbb{V}_i}{\partial \mathbf{q}} = \frac{\partial \mathbb{V}_i}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \mathbf{q}} = \frac{\partial \mathbb{V}_i}{\partial \mathbf{x}} \frac{\partial \phi_i(\mathbf{q})}{\partial \mathbf{q}} = \frac{\partial \mathbb{V}_i}{\partial \mathbf{x}} \mathbf{J_i}(\mathbf{q}).$$

Now, it remains to find the derivative of the GP variance. It is defined as

$$\begin{aligned} \frac{\partial \mathbb{V}_i}{\partial \mathbf{x}} &= \frac{\partial}{\partial \mathbf{x}} \left(k(\mathbf{x_i}, \mathbf{x_i}) - \mathbf{k}^T \mathbf{G^{-1}} \mathbf{k} \right) \\ &\text{$k(.,.)$ is stationary in our case and has 0 derivative} \\ &= -\frac{\partial \mathbf{k}^T \mathbf{G^{-1}} \mathbf{k}}{\partial \mathbf{x}} = -\frac{\partial \mathbf{k}^T \mathbf{G^{-1}} \mathbf{k}}{\partial \mathbf{k}} \frac{\partial \mathbf{k}}{\partial \mathbf{x}} \\ &= -2(\mathbf{k}^T \mathbf{G^{-1}}) \frac{\partial \mathbf{k}}{\partial \mathbf{x}} = -2(\mathbf{k}^T \mathbf{G^{-1}}) \dot{\mathbf{k}}. \end{aligned}$$

For the Gaussian covariance case we have derived $\mathbf{k}$ and $\dot{\mathbf{k}}$, so finally, for the i-th of the d points we get:

$$\frac{\partial \mathbf{y}_i}{\partial \mathbf{q}} = -2((\mathbf{k}^T \mathbf{G^{-1}}) \dot{\mathbf{k}}) \mathbf{J_i}. \tag{2}$$

With the so defined differentiable motion feature, we augmented the task space and can specify goals directly using our new Task Variable.

2) Control laws: The Jacobian of a task space feature is a direct prescription, what has to be done on infinitesimal scale in order to maximize/minimize the feature, i.e. a control law: Locally, the inverse Jacobian gives the $\Delta \mathbf{q}$ which fulfills the task.

The variance gradient points likely away from the surface. Thus, if we combine a variance task which favours uncertain regions with grasp tasks, their goals contradict, and we need to account for that by specifying relative weightings.

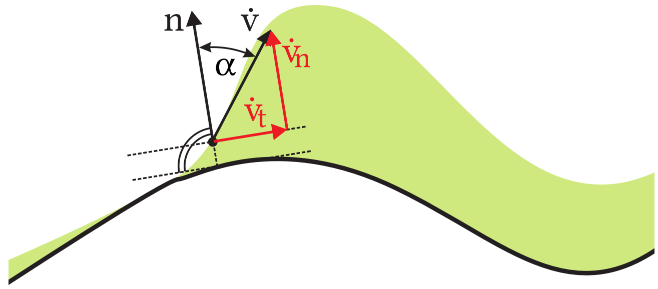


Fig. 1: Gradient of the GPISP variance split into components normal and tangential to the surface. The solid curve is the implicit surface, $\{\mathbf{x} : f(\mathbf{x}) = 0\}$; variance along the surface is depicted as shaded area; $\dot{\mathbf{v}}$ is the variance gradient and $\dot{\mathbf{v}}_n$,$\dot{\mathbf{v}}_t$ its components.

To make this trade-off more explicit we modify the above Jacobian to use the component of the variance gradient in the direction along the surface instead of the proper variance gradient. Figure 1 is a sketch of a variance gradient and its components at a low-variance point near surface. Assuming vectors $\mathbf{n}$ and $\dot{\mathbf{v}}$ to be known, $\mathbf{n} = \nabla\psi(\mathbf{x})$, $\dot{\mathbf{v}} = \nabla\mathbb{V}[\psi(\mathbf{x})]$, we can compute the direction component in tangential direction as $\dot{\mathbf{v}}_t = \dot{\mathbf{v}} - \dot{\mathbf{v}}_n$. The component in normal direction is

$$\dot{\mathbf{v}}_n = \frac{\mathbf{n}}{||\mathbf{n}||} ||\dot{\mathbf{v}}|| \cos(\alpha).$$

With the inner product $\mathbf{n} \circ \dot{\mathbf{v}} = ||\mathbf{n}|| ||\dot{\mathbf{v}}|| \cos(\alpha)$, we get

$$\dot{\mathbf{v}}_t = \dot{\mathbf{v}} - \left(\frac{\mathbf{n} \circ \dot{\mathbf{v}}}{||\mathbf{n}||^2} \right) \mathbf{n}.$$

Now, we can substitute this for the variance gradient in (2),

$$\frac{\partial \mathbf{y}_i}{\partial \mathbf{q}} = \left(\dot{\mathbf{v}} - \left(\frac{\mathbf{n} \circ \dot{\mathbf{v}}}{||\mathbf{n}||^2} \right) \mathbf{n} \right) \mathbf{J},$$

and obtain a new more precise control law which states *maximize variance, keeping constant distance to surface*. This formulation suits grasping context better, since it doesn't interfere with the natural task to establish contact with the object.

C. Exploration and exploitation

In the context of grasping with tactile sensors, to explore means to acquire information about all areas of a surface and create better model, while for exploitation one takes an existing model of an object and does one's best to grasp it. Other sensor modalities suggest different explore actions: in the presence of visual sensor, the effector may push an object to the side to avoid occlusion and explore visually the previously unseen region.

The uncertainty aware control laws from Section IV-B can be tuned to span the spectrum between pure exploration and pure exploitation and can be used as building blocks for explore-exploit grasp sequences. This makes repetitive grasping to an active learning setting, where each next interaction with an object can be viewed as a trade off between exploitation of the more certain regions of the estimate and exploration of unseen (or less certain) regions.

In order to define a policy for such grasp series, we need to decide what to trade off. We can support such decisions by local and global variance measures. Locally, the measured variance answers how sure we are that the suitable contact points for a grasp, according to the MAP estimate, are indeed so. Globally, the variance integrated along (what we believe to be) the surface tells overall, how far we can do better, i.e. would an explore step substantially gain information. Investigating complex policies and development of trade-off criteria for a meta controller goes beyond the scope of this paper. Here we focus on the usefulness of uncertainty aware control.

Consider for example the trivial policies *always explore* and *always exploit*. The former is expected to estimate effectively an unknown object and the latter to grasp the object robustly. Pure exploration demands a stopping criterion which can be again derived from the accumulated variance along the belief surface: for instance, if the decrease in variance flattens out, this could be a sign, that we know the surface good enough. Pure exploitation can go on until successful grasp. For repeating explore cases, in order to make less likely a case in which the same trajectory is repeated with no information gain a slight stochastic bias towards a random approach direction can be added.

One can also think of grasps which lie in between explore-grasp and exploit-grasp: a control law could be set up to assign explore task to one effector and exploit to other, for instance, one finger to place at a known region and its opposing finger to explore uncertain region.

D. Integration – an iterative explorative grasp policy

The blind touching scenario we began this text with amounts in terms of Section IV-C to a pure explore-grasp policy for iterative reactive grasping. In the following we describe an implementation of this policy integrated in a general framework for planning whole body grasp motion. We implement it for a Schunk Arm and attached Schunk Dexterous Hand in simulation.

1) Grasping sequence: The reactive grasping follows the schema lined out in Algorithm 1. For each grasp we specify

Algorithm 1 Reactive grasping

```
GP: add initial random observation
loop
    Q = plan(GP)
    (Q_p, contacts) = play(GP, real)
    if not contacts then
        (Q_p, contacts) = close()
    end if
    GP = update(contacts)
    backw(Q_p)
end loop
```

a kinematic motion trajectory in the configuration space: for given object model, robot kinematic chain and environment, we need to find a sequence of joint angles, $\mathbf{Q} = \{\mathbf{q_i}\}_{1:N}$, which starts in a known home position, $\mathbf{q}_1 = \mathbf{0}$ and ends in a feasible grasp configuration, while avoiding collisions with other objects and staying within the allowed joint limits. This postulates an optimisation problem, which is solved by $plan()$. Its definition is subject of the next Section IV-D.2. $play()$ follows the trajectory produced by $plan()$ in a feedback loop until a contact is registered, which provides an observation. If no contact is registered the fingers begin to $close()$ until contact with the object or self collision (when the plan or the model turn out to be bad). Finally, the arm plays the trajectory backwards and starts over with an updated model.

2) Trajectory optimisation: The forward trajectory is a result of optimisation problem which is constructed from several TVs implementing feasibility criteria and heuristics for hand and finger orientation to lead to a good grasp. A TV $y \in \mathbb{R}^d$ consists of

- update rule to compute the value for given configuration $\phi : \mathbb{R}^{|q|} \mapsto \mathbb{R}^d$,
- target value y^*,
- the partial derivatives in each q-component, $\mathbf{J} = \frac{\partial \phi}{\partial \mathbf{q}}$. The Jacobian is used for local linearisation around particular $\mathbf{q}$. Knowing $y = \phi(\mathbf{q})$, it tells us on an infinitesimal scale what direction to go from $\mathbf{q}$ for getting nearer to y^*.
- precision ρ, which states how important is to reach y^*.

Assume we have TVs $y_{1:n}$ with $y_i \in \mathbb{R}^{d_i} \forall i = 1..n$. We use them to design an objective function: we specify a target value y_i^*, as well as an weight of how important this target is relative to other variable's targets, ρ_i. The TVs introduce costs which arise from the difference between desired and present value. For a given configuration:

$$c(\mathbf{q}) = \sum_{1}^{n} \rho_i (y_i^* - \phi_i(\mathbf{q}))^2.$$

For each step of a discretized trajectory, for each task variable we come up with a target y_i^* and precision ρ_i. The sum over all time steps constitutes the total task cost of the trajectory: $C_t = \sum_1^T c(\mathbf{q_t})$. The minimisation of the total task costs is done by a modified Gauss-Newton method. The problem is a convex quadratic program for the optimal $d\mathbf{q}$ since the differentiable features are locally linearized with the Jacobian. For the configuration space we use a metric W which can be thought of as the covariance matrix of a Gaussian distribution for the state transition. The numbers on the diagonal of W imply the agility of the particular joint of the robot. Using $c_w W$ instead of W for some constant $c_w > 1$ makes the degrees of freedom less likely to move between time steps and counteracts increasing task precision ρ.

3) Tasks: The minimisation of the cost function leads to simultaneous satisfaction of the TVs according to the desired precision. Our TVs account for collision avoidance, joint limit avoidance, wrist orientation, finger positioning and orientation. Collision TVs work on proximities between meshes or point clouds; joints have hard-coded hardware specific upper and lower limits and a dedicated TV penalizes going near the limits directly in configuration space. Position

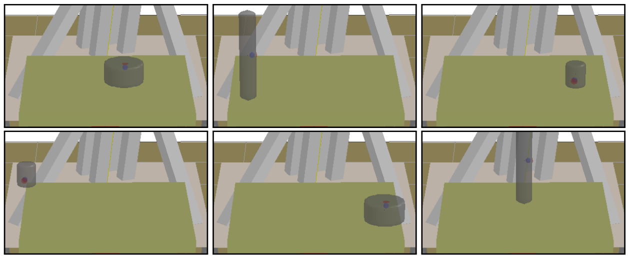


Fig. 2: Example environment samples used in the experiment: uniformly distributed position and object type. A red dot denote the initial random observation from the surface.

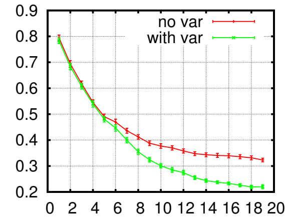
and orientation TVs are object centric – they are defined in terms of the GPISP model, rather than in Euclidean space, i.e. $\phi_i = \phi_i(\psi, \nabla\psi, ...)$. One of these task variables is defined as in Section IV-B.1 for the three finger tips of the Schunk Hand and implements the uncertainty awareness: $\phi_{\mathbb{V}} = (\mathbb{V}_{1:3})$. What Sections I-.1 and I-.2 suggest can be implemented here in the following way: conditioning $\phi^*_{\mathbb{V}} = \mathbf{0}$ tends to grasp better known regions, while $\phi^*_{\mathbb{V}} = \mathbb{V}_{max}$ tends to explore new regions. In the experiments we test the exploration behaviour and use the latter setting.

V. Experiments

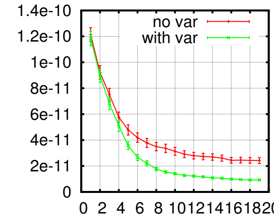
Using the controller described in Section IV-D we conduct experiments to illustrate the effects of uncertainty aware grasping. To this end, we place an invisible for the robot object within its work area, as in Figure 2. The position is uniformly random sampled in two dimensions, as is the appearance of the object among three predefined – lengthy, small and big one. Different positions on the table imply different kinematic constraints and ensure we test large region of the configuration space. The three different geometries account for objects which 1) are small and mostly fit in the hand of the robot; 2) are long and need to be explored along single dimension, possibly from single approach direction; 3) need to be explored from multiple approach directions. Latter two are expected to emphasize the benefit of our method.

In the experimental setting we assume a fixed object in reaching distance and provide an initial, very vague guess about the object – this is realized as a random observation from the object surface fed into the GPISP. It appears as approximately a small sphere at that point. The controller tries to grasp, relying only on the available model, and in its early attempts it is deemed to fail due to large discrepancy to the actual object shape: on its trajectory the robot arm collides with the real object. This sensory event provides function value and gradient observations for the GPISP and the model gets incrementally updated. After the collision, the robot arm goes back and starts a new attempt to grasp the updated model.

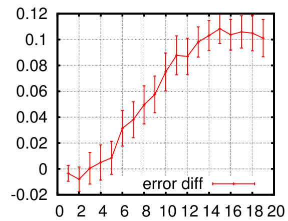
A major part of the repetitive grasping procedure is the model update upon collision. We assume the contact can be detected at any point of the robot body, which is feasible in simulation but needs to be taken care of with real world robots, since commonly tactile sensors are only available on very limited area of the body.



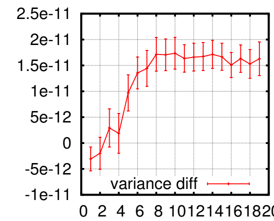
(a) surface approximation error



(b) variance along real surface



(c) difference in approximation



(d) difference of model variance

Fig. 4: Statistics collected over about 2x hundred runs (a run *with* and one *without* notion of uncertainty), each performing 20 attempts to grasp with incrementally updating model. Error bars show plus/minus one standard deviation of the estimator.

A. Results

The experimental results compare two controllers which employ very similar sets of TVs for trajectory optimisation; the only difference is that naïve control lacks the variance TV. For a random sample of the environment, i.e. the same object at same position with same initial observation, each controller is started once. Each run makes up to 20 successive grasps, incrementally updating its belief about the real object. Figure 3 shows a sequence of explore-grasps based on an updating GPISP estimate of a real object. The supplementary film from an example run provides an impression of the controller in action.

1) Measurements: On each grasp we record two statistics to compare the controllers:

- how far the approximation deviates from the real surface. The measure is defined as the common volume of true object and belief object, v_c, penalized by the overestimated volume, v_o, all normalized to the total volume of the real object v_t. This number, the second term in (3), is at most 1 but due to the overestimation term, it can drop below 0. Subtracting it from 1 gives the approximation error:

$$\epsilon = 1 - \frac{v_c - v_o}{v_t}. \tag{3}$$

- how certain the model is, measured on the surface of the true object, i.e. a numerical integration of the model variance along the true surface, S. In order to make the results comparable, here we again normalize the uncertainty by division by the volume of the voxels we visit on the surface:

$$\omega = \frac{1}{|S|} \int_S \mathbb{V}[v] \mathrm{d}v.$$

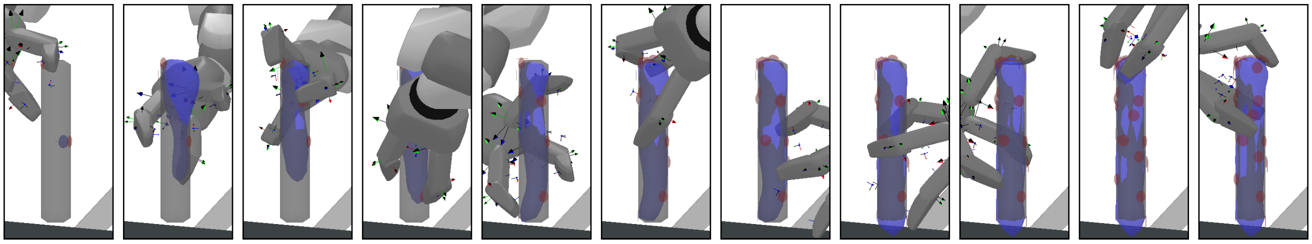


Fig. 3: An explorative grasp sequence. transparent grey: real object, unknown to the robot; blue: belief; red: contact points. Estimate after 2,4,5,6,8,11,13,15,16,19,20 grasps.

Figure 4a and 4b show the evolution of these statistics with every next grasp, averaged over multiple runs. However, they mask the identity of the two runs which correspond to a single environment sample. To make this correspondence more explicit, in Figure 4c and 4d we plot the average of the difference of approximation quality with and without notion of uncertainty and, respectively, the difference of model variance along the true surface.

The results give preference to the approach aware of model uncertainty. The estimation error is nearly identical for the first 4-5 grasps, but the naïve approach flattens out earlier. The added bias to grasp unknown regions leads to better estimates. In addition, it helps to reach same quality estimates much earlier: already after 10 grasps the augmented controller performs better than the other after 20. In the experiment, there is a table in front of the arm and the objects are placed above it. The collision avoidance task repels the hand from the table and makes it harder to explore the objects from below. This leads to either over or under estimation in the lower part, which explains why the estimation error doesn't drop below 0.2.

The average uncertainty on the real surface, ω, also supports the suggestion that the use of model variance in the control law observes the object in more informed way. After 6–7 grasps the one controller is as confident as the other after 20, i.e. even with comparable estimation error the variance control law has gained more valuable observation points.

We again refer to the film accompanying this paper for an illustration of the integrated approach.

VI. Summary

GPISP is capable of conveying sensor uncertainty to model uncertainty by transforming sensor (im)precision into variance of the GP posterior distribution. Building on this feature the present paper has following contributions:

- We derive a control law which offers a way to give preference to regions of the model with particular certainty level.
- In turn, by adding this to a grasp controller we introduce the notion of explore-grasp and exploit-grasp primitives – the former tending to grasp at unknown regions, the latter to grasp at better known points.
- Realisation of a policy which uses exclusively explore-grasps: we integrate the novel control law into a holistic framework for generating motion for repeating reactive grasps.

Under uncertainty coming from sensors or control, policies which trade off exploit and explore grasps can make grasping unknown objects more robust and improve the performance of tactile exploration.

We focus on the latter scenario and with a simulated Schunk Robot Arm and Schunk Dexterous Hand we conduct an experiment which shows the expected improvement – the developed control law increases the performance of tactile exploration, generalising well across objects of different sizes and positions.

Future research directions and topics include the realisation of similar scenarios on a real robot with limited tactile sensor area, and develop more sophisticated grasp-specific explore-vs-exploit policies.

References

[1] A. Bierbaum and M. Rambow, "Grasp affordances from multi-fingered tactile exploration using dynamic potential fields," in *Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on*. IEEE, 2009, pp. 168–174.

[2] J. Blinn, "A generalization of algebraic surface drawing," *ACM Transactions on Graphics (TOG)*, vol. 1, no. 3, pp. 235–256, 1982.

[3] S. Dragiev, M. Toussaint, and M. Gienger, "Gaussian Process Implicit Surface for Shape Estimation and Grasping," in *Robotics and Automation (ICRA), International Conference on*. IEEE, 2011. [Online]. Available: userpage.fu-berlin.de/mtoussai/publications

[4] K. Hsiao, L. Kaelbling, and T. Lozano-Pérez, "Task-Driven Tactile Exploration," 2010.

[5] H. Intraub, "Rethinking Scene Perception:: A Multisource Model," *Psychology of Learning and Motivation*, pp. 231–264, 2010.

[6] D. Katz and O. Brock, "Manipulating articulated objects with interactive perception," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, 2008, pp. 272–277.

[7] O. Kroemer, R. Detry, J. Piater, and J. Peters, "Combining active learning and reactive control for robot grasping," *Robotics and Autonomous Systems*, 2010.

[8] D. Meagher, "Geometric modeling using octree encoding," *Computer Graphics and Image Processing*, vol. 19, no. 2, pp. 129–147, 1982.

[9] C. Rasmussen and C. Williams, *Gaussian processes for machine learning*. Springer, 2006.

[10] A. Saxena, J. Driemeyer, and A. Ng, "Robotic grasping of novel objects using vision," *The International Journal of Robotics Research*, vol. 27, no. 2, p. 157, 2008.

[11] J. Steffen, R. Haschke, and H. Ritter, "Experience-based and tactile-driven dynamic grasp control," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*. IEEE, 2007, pp. 2938–2943.

[12] F. Steinke, B. Scholkopf, and V. Blanz, "Support vector machines for 3D shape processing," in *Computer Graphics Forum*, vol. 24, no. 3, 2005, pp. 285–294.

[13] O. Williams and A. Fitzgibbon, "Gaussian process implicit surfaces," *Gaussian Proc. in Practice*, 2007.

[14] H. Zha, T. Hoshide, and T. Hasegawa, "A recursive fitting-and-splitting algorithm for 3-D object modelingusing superquadrics," in *Fourteenth International Conference on Pattern Recognition, 1998. Proceedings*, vol. 1, 1998.