Entropy-Based Strategies for Physical Exploration of the Environment's Degrees of Freedom*

Stefan Otte[1] Johannes Kulick[1] Marc Toussaint[1] Oliver Brock[2]

***Abstract*— Physical exploration refers to the challenge of autonomously discovering and learning how to manipulate the environment's degrees of freedom (DOF)—by identifying promising points of interaction and pushing or pulling object parts to reveal DOF and their properties. Recent existing work focused on sub-problems like estimating DOF parameters from given data. Here, we address the integrated problem, focusing on the higher-level strategy to iteratively decide on the next exploration point before applying motion generation methods to execute the explorative action and data analysis methods to interpret the feedback. We propose to decide on exploration points based on the expected information gain, or change in entropy in the robot's current belief (uncertain knowledge) about the DOF. To this end, we first define how we represent such a belief. This requires dealing with the fact that the robot initially does not know which random variables (which DOF, and depending on their type, which DOF properties) actually exist. We then propose methods to estimate the expected information gain for an exploratory action. We analyze these strategies in simple environments and evaluate them in combination with full motion planning and data analysis in a physical simulation environment.**

I. Introduction

Most robotic tasks require manipulating objects in the world. Opening drawers to retrieve an object, grasping a door handle to open a door, or pressing a button to switch on the light—these are examples for tasks robots must be able to perform. All of these tasks involve the actuation of degrees of freedom (DOF) external to the robot. While learning about the internal DOF of the robot has long been a focus of system identification research [1], the problem of learning the DOF external to the robot has only recently received attention [2]. While there exists robust perceptual algorithms to perceive DOF from object movement [3], [4], the question of how to explore the environment to collect such data, find these DOF, and to acquire information about their characteristics, e.g., joint type or orientation of joint axis, has not been thoroughly studied.

We refer to this problem as the physical exploration challenge: autonomously discovering and learning how to manipulate the environment's degrees of freedom (DOF) by iteratively identifying promising points of interaction and trying to push or pull object parts to reveal DOF and their properties. To address this challenge, we combine research from active learning [5], bandits [6], and robotics to devise an approach to exploration of DOF in complex environments. Active learning strives to maximize the learning rate by actively choosing the most informative samples, whereas the theory of bandits deals with the problem of sequential decision making. Both of these aspects must be considered for explorative robot behavior. We leverage insights from these fields to formulate optimality criteria for exploration strategies. However, we first must formulate belief representations that capture the task-relevant structure of the environment in ways that enable us to explore external DOF efficiently.

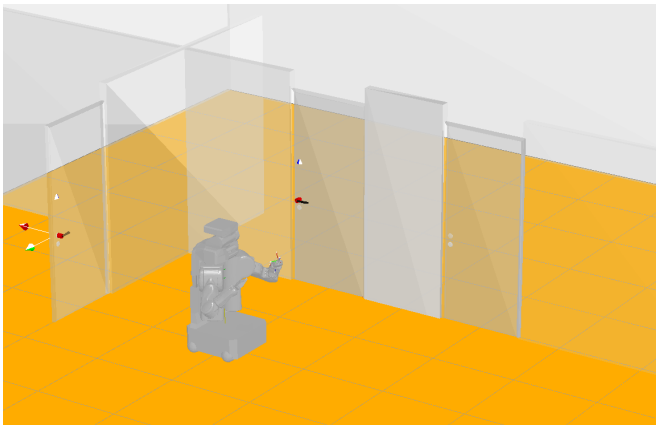

Fig. 1. The robot interacts with the environment and updates its belief in response to its observations. Transparency corresponds to the amount of knowledge available to the robot, where a fully transparent object indicates no knowledge and a solid object means full knowledge.

The following are our main contributions:

- We formulate the exploration challenge in complex environments with many DOF (see Sec. III).
- We propose a probabilistic belief representation to capture the robot's current knowledge state, including uncertainty about the environment's DOF, their properties, and relations (see Sec. IV).
- We propose a novel way to deal with the fact that the robot cannot initially know which random variables actually exist, as this depends on which DOF exist and their type.
- We propose how a consistent definition of expected entropy (or change of entropy) over our hybrid belief representation and leverage results from active learning to define various exploration strategies (see Sec. V).
- We integrate methods in a physical simulation envi-

*Johannes Kulick was funded by the German Science Foundation (DFG) under the priority programm "Autonomous Learning" (SPP 1527). Oliver Brock gratefully acknowledges financial support by the Alexander von Humboldt foundation through an Alexander von Humboldt professorship (funded by the German Federal Ministry of Education and Research).

[1] Machine Learning and Robotics Lab, Universität Stuttgart, Germany, [firstname.surname]@ipvs.uni-stuttgart.de

[2] Robotics and Biology Laboratory, Technische Universität Berlin, Germany, oliver.brock@tu-berlin.de

ronment including a simulated PR2, motion planning and execution methods, and data analysis methods to interpret the action effects. We use this to demonstrate autonomous exploration of complex environments with dynamics (see Sec. VII) and compare different exploration strategies.

II. Related Work

We discuss the state of the art in the following research areas, relevant to the exploration challenge: *A.* exploration and learning theory, *B.* belief representation, and *C.* interactive perceiving and learning of articulated objects.

A. Exploration and Learning Theory

Machine learning includes many formalisms for which exploration is important. In particular, active learning strategies maximize the expected learning rate, which can be expressed using Shannon information gain, model entropy, or the agreement within a committee of learners [5], [7], [8]. Recently, strategies based on upper confidence bounds (UCB) have been shown to posses a bounded sub-optimality in multi-armed bandit scenarios [6]. This has significantly advanced our understanding of exploration strategies, and has also advanced research in active learning [9]. Beyond active learning, exploration also plays a central role in reinforcement learning, where exploration strategies like E^3, Rmax, or the Bayesian Exploration Bonus have been proposed [10], [11], [12].

All of the above exploration strategies require a representation of uncertainty of the current world model, be it explicit or implicit. We believe that this is one of the core challenges when trying to transfer theoretically grounded exploration strategies from different areas of machine learning to real-world exploration scenarios: How can we efficiently represent a belief over the environment's DOF, their hybrid properties and relations? And how can we derive effective exploration strategies based on these representations? The standard approaches in active learning, bandits and RL lack belief representations of sufficient expressiveness to address the exploration challenge.

B. Belief Representation

A probabilistic belief representation allows to reason about the uncertainty of the belief and informs efficient exploration strategies. Graphical Models and Bayesian inference are widely used as probabilistic representations within machine learning and statistics [13]. Probabilistic belief representations are also well established in some subfields of robotics, including SLAM [14] and stochastic optimal control [15], [16]. However, in these areas, belief representations are mainly used for estimating the *own* state. In contrast, Kaelbling and Lozano-Pérez use a probabilistic belief representation to represent the pose and pose uncertainty of objects in the world [17]. They then reason and plan using this representation. However, their work does not consider uncertainty over the existence of DOF and their properties and relations. In more recent work, they explore various articulated mechanisms in a probabilistic representation. However, they focus on determining the joint type of single joints and do not consider the discovery of objects and joints in the environment [18]. To overcome this limitation, we will introduce a novel belief representation to enable hierarchical representation of uncertainty over DOF types, and, depending on the type, their properties and relationships.

C. Interactive Perceiving and Learning of Articulated Objects

Once an appropriate belief representation is identified, it needs to be filled in from sensory experience. In the context of the exploration challenge, this requires the identification of DOF and their parameters from observations. Over the last few years, the task of perceiving DOF from feature trajectories has been identified by Katz et al. [3]. In complementary work, Sturm et al. [4] proposed a method of identifying joint types from object trajectories. These methods show that it is possible to robustly identify the kinematic type of links from their movements. These movements, however, are scripted and the challenge of automatically generating such movement to perform exploration has only been explored superficially in a very simple setting [2].

Endres et al. [19] acknowledge that, in addition to the kinematic structure of the world, the dynamics of objects also play an important role. They learn the dynamics of doors through manipulation to be able to predict the behavior of the swinging door. The data used for learning is generated by the robot itself.

The exploration challenge is based on the idea that the robot can obtain perceptual information from its own interactions, instead of relying on the perception of static scenes. Katz et al. [20], [21] proposed a method based on this idea to clear a pile of objects. The image is segmented, one of several pre-defined actions is selected, and the object in the pile is either moved to confirm the segmentation or put into a basket. While these works clearly demonstrate the advantage of interactive perception, they do not address the full exploration challenge or discuss the exploration strategies on a theoretical level. This will be the focus of this paper.

Van Hoof et al. [22] apply an approach based on information-theory to select the predefined actions to explore the scene of objects. The scene is represented as graph. The decomposition of the scene into objects is learned through the "maximally informative interactions". However, they do not consider DOF in the environment.

All papers discussed so far describe learning from observations, an essential subproblem of the exploration challenge. Our contribution lies in the exploration strategy as a whole, including action selection based on expected information gain, criteria to decide on the next manipulation action, and autonomously generating motions that are informative about the properties of DOF properties and their relations, including joint type, friction and inertia, and joint limits.

III. Problem Statement (alternative)

Definition: We define the *exploration challenge* as the task in which an agent has to minimize the uncertainty about the environment. It can actively select objects in the environment, interact with them and learn their properties. More formally, we define the objective of the exploration challenge to efficiently minimize the total sum of the entropy over properties θ^i of object i

$$\min \sum_i H(\theta^i). \tag{1}$$

This definition of the exploration challenge is deliberately general to make it interesting in the long run. To tackle this problem in this paper, we make some assumptions about the world and further limit the number of properties we are interested in.

We assume the environment is a *rigid-body world*, i.e., the environment consists of rigid objects, which can be static, freely movable, or connected through joints (such as rotational or prismatic joints). A rigid-body world can contain walls, doors, windows, cabinets, drawers, any articulated tools, etc. Also, our rigid-body worlds are deterministic, i.e., object and joint parameters, like mass or joint limits, remain constant. The DOF of the environment are thus defined by the joint configurations of freely movable objects. However, all the robot's observations are noisy.

The rigid-body assumption excludes worlds containing, for example, liquids and flexible objects, but captures a wide range of possible and interesting worlds. These worlds serve as an adequate testbed for the exploration challenge and facilitate the usual perception, manipulation, and planning tasks in robotics.

For this paper, we also assume that the robot has access to a segmentation of the scene into rigid bodies (objects); the robot can perceive (without physical interaction) their shapes and poses, but the kinematic structure and the dynamic properties of the joints are initially unknown.

We propose the following minimum set of world properties a robot should know about to be able to accomplish generic manipulation tasks in a rigid-body world:

- *Object type:* An object can be either movable or static.
- *Joint type:* If there exists a joint between two rigid bodies, the joint can be either *rotational* or *prismatic*.
- *Joint limits:* Each joint has a mechanical limit, captured by a minimum and maximum value of the joint variable.
- *Friction coefficient:* Friction slows down the movement of a shape attached to a joint. Although different physical things act together (friction forces, inertia etc.), we summarize them in a single parameter.

We therefore define random variables for each property of an object i: object type O^i, the joint type J^i, and we group the joint properties—upper and lower joint limit and friction coefficient—in one parameter RV for each joint type (θ_r^i for the rotational and θ_p^i for the prismatic joint).

Note that in this setting choosing an object to explore is similar to choosing a bandit to play in the bandit scenario.

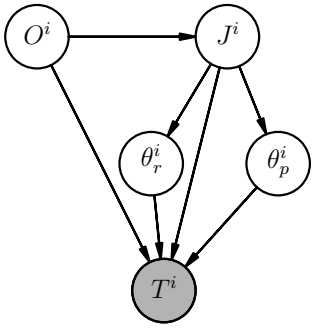

Fig. 2. Graphical model that represents the belief associated to a single object i. Object type O^i, joint type J^i, joint parameters for the rotational joint θ_r^i, joint parameters for the prismatic joint θ_p^i, the observed trajectory T^i of the object.

IV. Belief Representation

We define a structured belief over the properties for each object (see Fig. 2). Each object in the scene graph is augmented with such a joint distribution. Fig. 1 shows a visualization of the scene graph which is augmented with the belief. The transparency indicates the entropy of the objects.

Let the index i denote the object i. In the following i is omitted for readability. Let O and J be discrete and θ_r and θ_p be a set of continuous random variables. The *object type* O indicates if the object is *movable* or *static*. Its discrete probability distribution is calculated from the number of times an object is observed to be *movable* or *static*. J represents the *joint type* which can be *rotational* or *prismatic*. To represent the dependent nature of the world, we also add a pseudo value *nil*, which we will explain in the next section. The probability distribution is estimated by counting the number of observations for each value. θ_r and θ_p are the parameters of the joints, consisting of upper and lower joint limit and a friction factor. These variables are assumed to be Gaussian distributed. Dependent on all theses variables is the trajectory T of the object. This variable is observable.

A. The nil Value

In our scenario the *existence* of random variables depends on other random variables, e.g., there is no joint type if the object is static. The discovery of a movable object, for example, would change the dimensionality of the belief and also abruptly change the entropy of the belief.

To avoid a trans-dimensional space we propose the following: We introduce a pseudo value for distributions which we call *nil* value. It captures the situation when the existence of the variable is unknown. Consider the case of object type O and joint type J. If we have a probability $P(O = \text{static}) = 0.3$, we only have a 70% belief that there is a joint type involved in the model. So we have a deterministic dependency between O and J, stating that the marginal is $P(J = \text{nil}) = 0.3$. We call J *nil-dependent* on O being

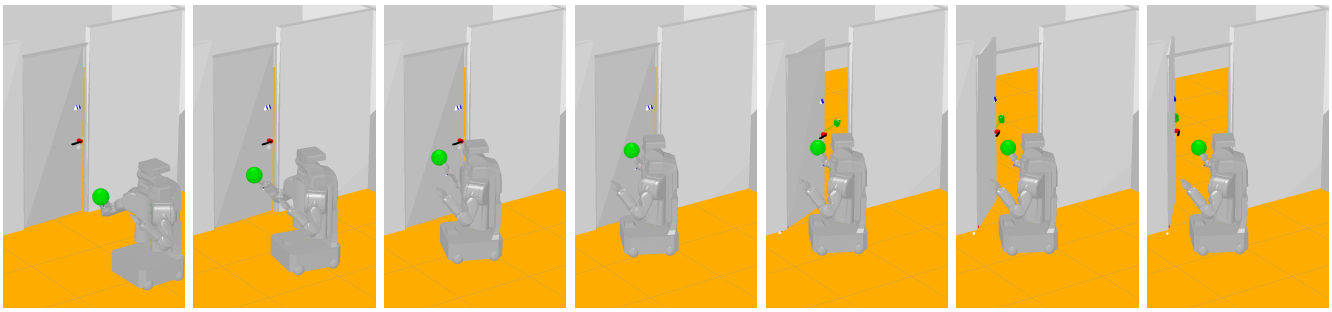

Fig. 3. To generate interactions with the world, we either use a 'flying ball' actuator (depicted as a green ball) or a simulated PR2 robot. Motions for both are generated with an RRT initialized trajectory which is then optimized. Thus both abstractions are able to generate very similar behavior. Here we show an interaction with a door for both types of interactions.

static. To avoid corrupting the entropy by possible existence or non-existence of a RV we have to handle the nil value case for discrete and continuous RV separately.

Definition: If X is a discrete random variable *nil-dependent* on another discrete random variable Y having the value v the conditional probability distribution is

$$P(X = \text{nil}|Y) = \begin{cases} 1 \text{ if } Y = v \\ 0 \text{ else} \end{cases} . \tag{2}$$

With these pseudo values, we can compute meaningful entropies over the distributions. Using this formalization, we can ensure that an information gain is associated with discovering that a static object does not have a joint, namely the information gain that results from the change of the variable from nil to the specific joint type.

The same problem arises with continuous distributions, but it is not possible to inject a pseudo-value here. We therefore use a Dirac delta as pseudo-value. The Dirac delta function has the minimal entropy (i.e. $-\infty$). Intuitively it can be thought of as an infinitesimal *nil* value. The Dirac delta simply formalizes in a continuous space that everything is known about a variable and no uncertainty is in play. Thus if we would know that a random variable does not exist, we could set its distribution to a Dirac delta, because there is no uncertainty over non-existing random variables. The only question would be, where to position the Dirac delta, since every actual value makes equally little sense—a non-existing variable does not have any value. We could introduce a a pseudo *nil* position, but it is unclear where this position should be and how a marginal over the existence of the random variable then would be computed. But since we use this pseudo value only for computing the entropy, the actual position of the Dirac delta is not important as long as it does not bias the marginal distribution. Since the entropy of the nil-depended distribution should decrease with increasing probability $P(Y = v)$ we center the Dirac delta at the mean of the nil-dependent distribution conditioned on $Y \neq v$. Thus we do not introduce any bias by the pseudo value.

Definition: If X is a continuous random variable nil-dependend on a discrete random variable Y, and v is a possible value of Y, the conditional probability distribution is given by:

$$p(X|Y = v) = \delta_{\mathrm{E}[p(X|Y \neq v)]}(X), \tag{3}$$

with $\delta_x(\cdot)$ being the Dirac delta function at position x and $\mathrm{E}[\cdot]$ the expectation.

Although this is the clear translation of the discrete to the continuous case, it is not possible to calculate the marginal of a random variable if a Dirac delta function is involved. We therefore approximate the Dirac with a very narrow Gaussian distribution. This is particularly useful since the joint parameters naturally have Gaussian distributions and the entropies become easily comparable.

B. Calculating the Entropy

Now, for each object i and its parameters $O^i, J^i, \theta^i_p, \theta^i_r$ we could calculate the entropy assuming independence of its parameters

$$H(O^i, J^i, \theta^i_r, \theta^i_p) = H(O^i) + H(J^i) + H(\theta^i_p) + H(\theta^i_r).$$

However, the distributions are not independent. We define an entropy measure, $\hat{H}$, which weights the entropy of each distribution according to its likelihood and takes the nil-dependence into account:

$$\begin{aligned} \hat{H}(O^i, J^i, \theta^i_r, \theta^i_p) &= H(O^i) && (4) \\ &+ P(O^i = \text{movable})\ H(J^i) && (5) \\ &+ P(J^i = \text{prismatic})\ H(\theta^i_p) && (6) \\ &+ P(J^i = \text{rotational})\ H(\theta^i_r). && (7) \end{aligned}$$

Note that $\hat{H}$ can be computed analytically because each distribution is either categorical or Gaussian.

V. Strategies

We compare four exploration strategies. The task of the strategy is to select an object to explore. The action that is performed is a simple "push the object" action. This is implemented as a fixed force applied to the object at a random surface point towards the center of mass of the object. The set of actions is therefore equal to the set of objects and the terms are used interchangeably.

1) **Random:** The agent chooses one of the objects in the world uniformly at random. This simple strategy serves as a baseline for our evaluation.

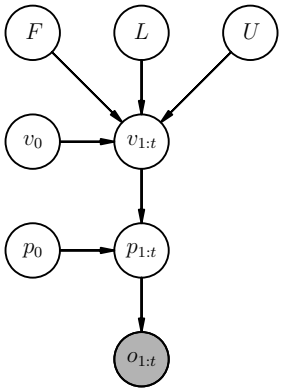

Fig. 4. Graphical model to learn the properties of a 1D point mass.

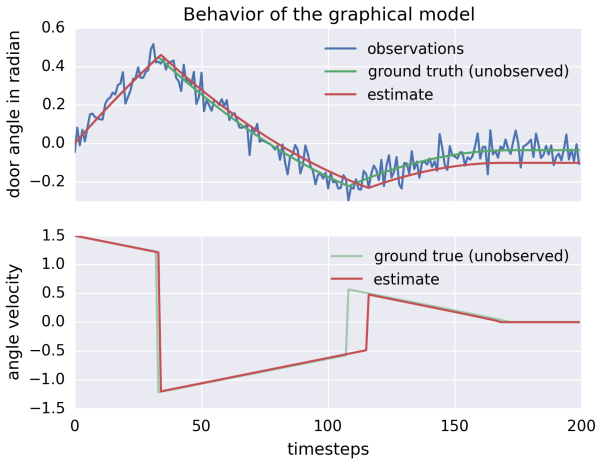

Fig. 5. After drawing 20,000 samples (burn-in phase: 5,000 samples, thinning: 2) and adjusting the parameters of the 1D point mass model to the MAP values, the graphical model predicts the shown movement of a door (red curve). The blue curve shows the observed data, whereas the green curve shows the (unobservable) ground truth.

2) **Round robin:** The agent selects objects sequentially. Although this strategy seems to be a very simple, one should note that for certain worlds—such as worlds that only consist of the same objects and therefore return the same reward/reduction of entropy—the round robin strategy yields optimal results.
3) **Expected change of entropy:** The agent computes the expected change of entropy for each object in the belief and then chooses the object that minimizes this criterion. Formally this means

$$i^* = \underset{i}{\operatorname{argmax}}\, \hat{H}(O^i, J^i, \theta_r^i, \theta_p^i) - E\left[\hat{H}(O^i, J^i, \theta_r^i, \theta_p^i \mid T^i)\right]. \quad (8)$$

Information-theoretic criteria are successfully applied in active learning scenarios and are promising candidates for the exploration challenge. However, calculating the expected change of entropy involves the computation of the expectation over all actions. This requires a forward model and is often expensive to compute or even intractable. We only have a finite set of actions and employ a simplified forward model to calculate the expectation of all actions. Updating all distributions can be done analytically and therefore calculating the expectation can be done analytically.
4) **Max Entropy:** The agent computes the current entropy for each object and chooses the one with the highest entropy.

$$i^* = \underset{i}{\operatorname{argmax}}\, \hat{H}(O^i, J^i, \theta_r^i, \theta_p^i) \quad (9)$$

The assumption is that objects with high entropy are not yet modeled properly and thus a large reduction in entropy can be expected from exploring it. This heuristic is successful in many kinds of problems, however, it pays unjustified attention to actions with random outcome.

VI. Motion Generation and DOF Parameter Estimation from Perception

A. Motion Generation

For both abstraction levels of our agent, the PR2 and the flying ball, we generate the full body motions with a rapidly-exploring random tree (RRT) [23]. It is a bi-directional RRT that samples directly in the joint space. To check whether the sampled joint positions are collision-free we use the SWIFT collision detection library[1]. This initial trajectory, which is normally not smooth, is fed into a operational space controller that optimizes the trajectory with a standard Newton optimization method. The resulting trajectory is then used to set the joint values of the agent directly. Fig. 3 shows an example of the generated motion for both abstraction levels of the agent.

B. DOF Parameter Estimation

The agent can perceive the position of objects in the world. Once it interacts with an object it perceives a 3D trajectory of the object. The object type O^i is updated with a *static* observation if there was no movement, or with a *moving* observation if movement was observed. For learning the joint type J^i and pose, we use the *articulation library* by Sturm [4]. The distribution for the joint type J^i is updated accordingly.

Because the articulation library does not infer joint properties such as joint limits and a friction coefficient, we employ a graphical model (shown in Fig. 4) to infer these parameters. First, we project the 3D trajectory into the space of the joint, i.e., the rotation about a revolute joint or the translation along a prismatic axis. The resulting 1D trajectory $o_{1:t}$ is derived from the underlying trajectory $p_{1:t}$ including

[1] http://gamma.cs.unc.edu/SWIFT

Gaussian noise ϵ:

$$o_{1:t} = p_{1:t} + \epsilon \tag{10}$$

where $\epsilon \sim \mathcal{N}(0, \sigma)$. The behavior of p_i can be described by the following process, which is derived from the physical equations of motion of a 1D point mass. Given the initial position p_0 and initial velocity v_0, the position and velocity can be predicted for each time step:

$$v_{t+1} = d_{t+1} \cdot \sqrt{v_t^2 - (2 \cdot F \cdot v_t \cdot \tau)}, \tag{11}$$

$$p_{t+1} = p_t + v_t, \tag{12}$$

where τ is the time difference between time steps. The factor d_t accounts for the change in the moving direction of a joint reaching the limit. The assumption of a perfectly elastic collision at the upper joint limits U and lower joint limits L results in

$$d_t = \begin{cases} 1 & \text{if } L < p_t < U \\ -1 & \text{if } p_t \leq L \text{ or } p_t \geq U \end{cases}. \tag{13}$$

We assume Gaussian priors around the initial position p_0 and velocity v_0, estimated by the observations. We choose a uniform prior for the friction coefficient F and the limits L and U.

To infer the parameters F, L and U which were summarized as θ_r and θ_p in the previous sections we use *PyMC* [24] —a Markov chain Monte-Carlo library— and in particular the Metropolis-Hastings algorithm [25] to approximate the distributions over these parameters. In following interaction steps the posterior is used as a prior for the sampling process.

Fig. 5 shows the typical behavior of this graphical model when observing a door for the first time. Consecutive observation of the same door further improves the learned model.

VII. Experiments

A. Toy World Experiment

Our first set of experiments are in a purely synthetic scenario to show the basic characteristics of a set of strategies. A set of two objects is given. One of them is static, one of them is attached to the static world by a prismatic joint. This is the minimal example, that can lead to interesting behavior. We expect an agent to focus on the movable object and ignore the static object after only very few interactions. In the scenario the agent—a purely algorithmic one with no physical representation—choses an object to explore. The agent then observes whether the object is static or movable, which type of joint it is attached to (if at all), and the value of the continuous parameters of the joint.

In Fig. 6, we show the situation after exploring each object five times. We show both discrete distributions along with their entropies and the expected change of entropy. The first two plots show the distributions of object type and joint type for each object, the third and forth show the discrete and differential entropies for each distribution and the last plot shows the expected change of entropy.

One can see that the first object is most likely a wall or other fixed object and the second object is movable based on a prismatic joint. Although both objects have been explored the same amount of times, the expected change of entropy is *higher* for the movable object. This is due to the fact that the static object with high probability has a non-existent joint type and joint properties (a nil value). Thus the entropy of those distributions is very small (see the third and fourth plot). Furthermore, the probability of change is small, since we are already certain of the object being static. Consequently, the expected change of entropy is also small.

An explorer maximizing the expected change of entropy or choosing the object with maximal entropy would choose the object with the prismatic joint over the static object. This is an interesting behavioral observation. Intuitively, it is a reasonable decision, since drawers and doors seem more interesting to us. With our belief representation we were able to catch this intuition by formally deducing higher Shannon information from those objects with larger parameter sets.

Fig. 7 provides further support for this statement. It shows the reduction of entropy achieved by the different strategies. The information theoretic driven strategies (expected change of entropy and max entropy) outperform the heuristic strategies, although the difference in such a simple scenario is small.

B. Physical Simulation

To test and compare the different strategies in a more realistic scenario, we set up a simulation of an environment with several DOF (see Fig. 1). In the abstract scenario, the agent is a "flying ball" (as an abstraction of a general end-effector), which can move freely through the space and can interact with the various objects. In the second scenario, the agent is a PR2 robot with two 7-DOF arms, a telescopic spine, two grippers and a omni-directional base. Both levels of abstraction are depicted in Fig. 3.

As shown in Fig. 8, in a more realistic scenario the difference between strategies is more pronounced. Although our observations are noise-free, we still have various sources of uncertainty. The physics simulation is not very precise and leads to unrealistic movements. Our 1D point mass model may not capture all of these effects. Also the joint pose inferred by articulation introduces a source of noise.

However, we can see that the strategies driven from information theory lead to better and faster uncertainty reduction. Also, the round robin strategy is still very successful. Since we investigate the complete model, this surprising result is reasonable. We have no specific task but to learn a precise model of the world. So the properties of *all* objects are equally important. Thus only the fact that static objects lack certain properties makes them less interesting. Still, each exploration leads to more certainty that they are static and thus to a reduction of uncertainty.

VIII. Conclusion and Future Work

We introduced the robotics exploration challenge. In this challenge, a robot is expected to learn as efficiently as possible about the degrees of freedom present in the environment. Pursuing such a challenge can advance the state of the art in

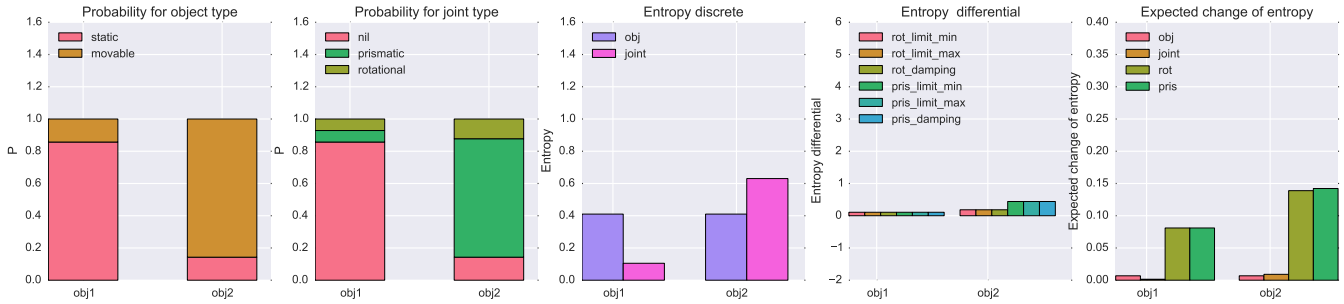

Fig. 6. An example of a belief after ten exploration steps: Both objects have been explored five times. The first two plots show the distribution of the object type and the joint type, the third and forth plot show the entropies of the various distributions—either discrete or continuous—and the last plot shows the expected change of entropy of each distribution.

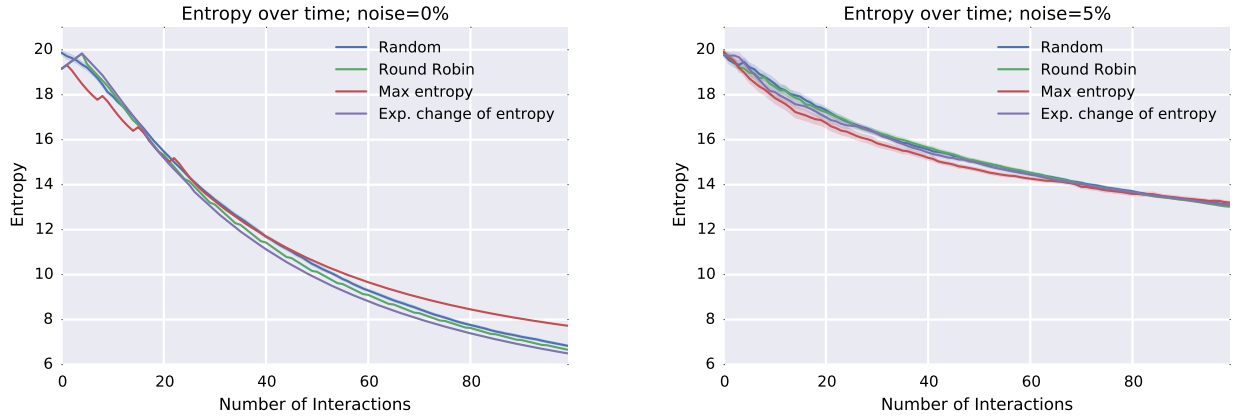

Fig. 7. The performance in minimizing the belief entropy of different strategies in the toy world (without noise and with 5% noise). 20 runs were performed.

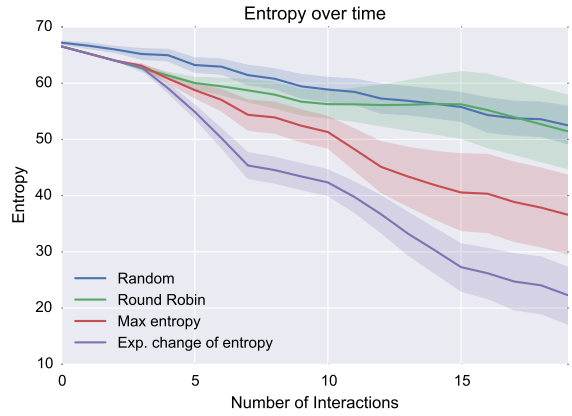

Fig. 8. The performance in minimizing the belief entropy of different strategies in the physical simulation experiment. 19 runs were performed.

robotics in two ways: first, addressing this challenge requires the integration of many components, including an adequate representation of information about the world, perceiving and acting in the world, and performing efficient exploration. While these components are usually developed and evaluated in isolation, we believe that the progress in each of the individual component can only be assessed in the context of an integrated system. Second, by enabling a robot to autonomously and effectively explore the world to achieve a learning objective, we can lay the representational and algorithmic foundation for autonomous skill acquisition in robots.

Towards these goals, this paper introduced a novel way of representing the world state, capable of handling uncertainty about the existence of objects. Based on this representation, we defined information theoretic exploration strategies. Our experimental evaluation shows that simple exploration strategies such as “random” and “round robin” show surprisingly good results in the toy world. However, the behavior created by information theory-based exploration strategies seems more plausible: a newly discovered articulated object is more interesting than a wall and invites further exploration before it becomes boring.

In the more complex physical simulation, this behavior also is able to explore a scene faster than the heuristics. This suggests that for complex scenarios, strategies derived from information theory and active learning will lead to better performance. Further research is needed to strengthen support for this statement.

Also, our research has its limitations. The actions the robot can choose are limited and most properties of an object can be learned reasonably well with only one interaction. In more

complex worlds, e.g., in worlds with coupled joints such as doors with handles and locks, smarter exploration strategies might be needed. Our hypothesis is that information theory-based strategies could lead to more interesting behaviors and better exploration in these kinds of scenarios. The "expected change of entropy" exploration strategy already showed promising results. This suggests that in complex worlds, techniques from active learning could also yield better results than naive and heuristic strategies.

The belief representation proved to be an important aspect of the exploration challenge—especially the dependency of the different random variables and the resulting entropy measure—and should be investigated further. We showed that comparing discrete and differential entropy is not straight forward. Although in principle they are comparable, in practice continuous distributions can carry more information than discrete distributions. Thus we need a theory which can handle this discrepancy.

We also want to further investigate how exploration strategies might be task-driven, i.e., only the DOF important for a specific task are explored, whereas irrelevant DOF should be ignored. This also raises the question of how knowledge from former exploration challenges can be translated during new exploration challenges. This includes both transferring knowledge between similar objects—doors, for example, are very similar most of the time—as well as leveraging experience to improve the exploration strategies themselves.

REFERENCES

[1] S. Khemaissia and A. S. Morris, "Nonlinear robot system identification based on neural network models," in *Int. Conf. on Intelligent Systems Engineering*, 1992, pp. 299 – 303.

[2] D. Katz, Y. Pyuro, and O. Brock, "Learning to manipulate articulated objects in unstructured environments using a grounded relational representation," in *Proc. of the Int. Conf. on Robotics: Science and Systems*, 2008, pp. 254–261.

[3] D. Katz, A. Orthey, and O. Brock, "Interactive perception of articulated objects," in *Experimental Robotics*, ser. Springer Tracts in Advanced Robotics, O. Khatib and G. S. V. Kumar, Eds. Springer, 2014, vol. 79, pp. 301–315.

[4] J. Sturm, C. Stachniss, and W. Burgard, "A Probabilistic Framework for Learning Kinematic Models of Articulated Objects," *Journal of Artificial Intelligence Research (JAIR)*, vol. 41, no. 2, pp. 477–526, 2011.

[5] B. Settles, "Active learning literature survey," University of Wisconsin–Madison, Computer Sciences Technical Report 1648, 2010.

[6] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning Journal*, vol. 47, no. 2-3, pp. 235–256, 2002.

[7] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proc. of the Annual Conf. on Computational Learning Theory*, 1992, pp. 287–294.

[8] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *Journal of Artificial Intelligence Research (JAIR)*, vol. 4, no. 1, pp. 129–145, 1996.

[9] N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger, "Information-theoretic regret bounds for gaussian process optimization in the bandit setting," *Transactions on Information Theory*, vol. 58, no. 5, pp. 3250–3265, 2012.

[10] M. Kearns and S. Singh, "Near-optimal reinforcement learning in polynomial time," *Machine Learning Journal*, vol. 49, no. 2-3, pp. 209–232, 2002.

[11] R. I. Brafman and M. Tennenholtz, "R-max - a general polynomial time algorithm for near-optimal reinforcement learning," *Journal of Machine Learning Research (JMLR)*, vol. 3, pp. 213–231, 2002.

[12] J. Z. Kolter and A. Ng, "Near-Bayesian exploration in polynomial time," in *Proc. of the Int. Conf. on Machine Learning (ICML)*, 2009, pp. 513–520.

[13] D. Barber, *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[14] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.

[15] R. F. Stengel, *Optimal control and estimation*. Dover publications, 1986.

[16] M. Toussaint, "Robot trajectory optimization using approximate inference," in *Proc. of the Int. Conf. on Machine Learning (ICML)*, 2009.

[17] L. P. Kaelbling and T. Lozano-Pérez, "Integrated Task and Motion Planning in Belief Space," *Int. Journal of Robotics Research*, vol. 32, 2013.

[18] P. R. Barragán, L. P. Kaelbling, and T. Lozano-Pérez, "Interactive Bayesian Identification of Kinematic Mechanisms," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2014.

[19] F. Endres, J. Trinkle, and W. Burgard, "Learning the Dynamics of Doors for Robotic Menipulation," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013, pp. 3543–3549.

[20] D. Katz, M. Kazemi, J. A. Bagnell, and A. Stentz, "Clearing a pile of unknown objects using interactive perception," DTIC Document, Tech. Rep., 2012.

[21] D. Katz, A. Venkatraman, M. Kazemi, J. A. Bagnell, and A. Stentz, "Perceiving, learning, and exploiting object affordances for autonomous pile manipulation," in *Proceedings of Robotics: Science and Systems*, Berlin, Germany, 2013.

[22] H. van Hoof, O. Kroemer, H. Ben Amor, and J. Peters, "Maximally informative interaction learning for scene exploration," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 5152–5158.

[23] S. M. LaValle and J. J. Kuffner Jr, "Rapidly-exploring random trees: Progress and prospects," in *In Proceedings Workshop on the Algorithmic Foundations of Robotics*, 2000.

[24] A. Patil, D. Huard, and C. J. Fonnesbeck, "PyMC: Bayesian stochastic modelling in Python," *Journal of Statistical Software*, vol. 35, no. 4, p. 1, 2010.

[25] S. Chib and E. Greenberg, "Understanding the Metropolis-Hastings Algorithm," *American Statistics*, vol. 49, no. 4, pp. 327 – 335, 1995.