Acting on Push Affordances: Adapting Dynamic Movement Primitives Based on Object Behaviour

Senka Krivic, Emre Ugur and Justus Piater

Abstract— *Nonprehensile* **manipulation such as pushing can play a significant role in complex scenarios. Objects may have diverse, even anisotropic properties under pushing in different environments. This increases the complexity of the pushing problem. We propose an approach to adapting dynamic movement primitives (DMPs) based on the observed object-motion behaviour and experienced forces. We also investigate an alternative optimal control based technique that enable dexterous and adaptive manipulation using pushability and manipulability of objects.**

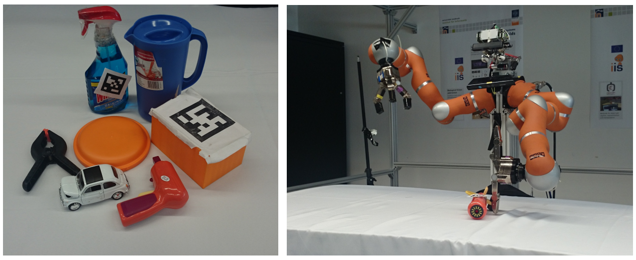

Fig. 1. Object data set used for pushing tests (left); pushing experiments with a KUKA LWR arm (right)

I. INTRODUCTION

Pushing is often used to simplify or to improve the precision of complex manipulative skills. If you imagine you are trying to place an object, for example to put a book on a shelf, or to carry out any tabletop manipulation, in most cases you will place the object approximately, and then use pushing movements to bring it to the desired position and orientation. Also, pushing is commonly used to move objects out of the way or to make grasping other objects easier. Enabling similar pushing skills for robots can serve multiple purposes: correcting the placement of an object, clearing paths and shoving objects into free space, maneuvering large objects or objects that are hard to grasp, enhancing pick-and-place operations, stc.

In our preliminary experiments, we observed that single-contact pushing of objects even in a straight line is not trivial due to differing properties of objects. Humans are able to push various objects in a desired way by pushing them approximately into designated directions and changing the hand motion based on observed senses and behaviours. We aim to incorporate a similar concept into the robot manipulation skills by adapting robot motion controllers based on the object behaviour and experienced forces. For motion control we use dynamic movement primitives (DMPs) which were proposed as an efficient way for learning and controlling complex robot behaviours [1].

*The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 (Specific Programme Cooperation, Theme 3, Information and Communication Technologies) under grant agreement no. 610532, Squirrel

All authors are with Institute of Computer Science, University of Innsbruck, 6020 Innsbruck, Austria, Contact information: senka.krivic@uibk.ac.at, emre.ugur@uibk.ac.at, justus.piater@uibk.ac.at

II. RELATED WORK

Many researchers have studied nonprehensile manipulation by pushing with robotic manipulators or with the robot base. This topic raised interest in different research areas which resulted in a variety of approaches and problems related to pushing. Early work on pushing done by Mason and Lynch [2] presented a theoretical model of the dynamics of pushing with the robotic manipulator. One of the first systems that took into account vision feedback for building a forward model of a planar object was created by Salganicoff et al. [3]. In later work, the action of pushing was often used for learning dynamic models, behaviours and object specific properties by observing results of the push [4], [5]. Several methods treat the problem of learning contact locations for successful pushes [6], [7]. Most of them focus on extracting the shape of the object and determining contact locations that are good for pushing.

Existing algorithms have several limitations which include the following. *(i)* Restriction to objects and environments which possess specific properties such as quasi-statically sliding on a high-friction surface. *(ii)* Pushing planners are usually based on geometric properties of objects and designed for specific scenarios. *(iii)* Pushing to the goal position is done in clutter-free environments. *(iv)* Many methods require a time-expensive learning phase depending on the object shape.

III. PROPOSED APPROACH

Objects may have diverse properties under pushing in complex scenarios. If you imagine a task, such as cleaning a child's room, a robot has to be able to maneuver diverse objects with different masses and friction distributions. In tests of simple, straight-line pushes which were done for a set of various objects (see Fig. 1), we observed that object loss is mostly dependent on the friction force and the object behaviour dynamics. Different object properties are manifested under different pushing dynamics and environment conditions. For example, objects can flip due to high friction or fast movements. In our pushing task, the robot is assumed to have no previous knowledge of the object. By pushing the object, it is possible to build an object behaviour model *on the fly* based on observations of object-environment dynamics. This model is then used to adjust the robot motion controller. The proposed framework is shown in Fig. 2..

A. Adjusting DMPs

Choosing suitable control parameter values for pushing is frequently object-dependent. Instead of tuning control parameters with respect to the object class or the shape, we propose identifying the object behaviour by sensing and introducing corrective actions in a pushing task. At the onset of a manipulation, the object is assumed to have ideal properties for pushing, moving only together with the robot. During the execution these assumptions are updated based on the object displacement or end-effector force values with respect to the dynamics of the robot movements. Control parameters are adapted based on the pushing experience to provide desired object movements. Online modulation of dynamic movement primitives is done by adding coupling terms. These terms alter a primitive based on intermediate success of object behavior and experienced forces:

$$\begin{aligned} \tau \dot{z} &= \alpha_z(\beta_z(g-y)-z)+f(x)+I_O \\ \tau \dot{y} = z\tau, &\qquad \dot{x} = \frac{-\alpha_x}{1+\alpha_e J_e} \end{aligned} \tag{1}$$

where:

$$\begin{aligned} I_O &= K_{tp}p(E\left[x_O\right]-x_O)+K_{tF}f(E\left[F|e_{\min}\right]-F) \\ J_e &= \int_{t_0}^{t_c} d\left(p_d\left(\gamma\right),x_O\right)\mathrm{d}t \end{aligned}$$

$K_{tp} = f(\mathrm{var}(x_O), E(x_O))$ and $K_{tF} = f(\nabla F, F)$ are time-varying task sensitivities with respect to object pose x_O and force F.

B. Optimal Control

As an alternative method, we propose an *optimal control* strategy with respect to the pushing objectives. Objectives which define the cost function are as follows: *(i)* The object should be delivered to the target pose as fast as possible within given tracking-accuracy constraints; *(ii)* Non-smooth and jerky movements of the robot should be minimized:

$$J = c_t\int_{t_0}^{t_f} t\ \mathrm{d}t + c_j\int_{t_0}^{t_f}\sum_{i=1}^{N}\left(\frac{\mathrm{d}^3x_i}{\mathrm{d}t^3}\right)^2\mathrm{d}t \tag{2}$$

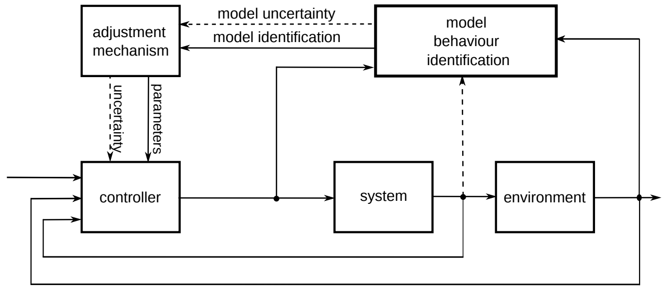

Fig. 2. The proposed model behaviour identification adaptive control approach for push-manipulation

Here, t_f is a free variable representing the final point in time, and jerk is defined in Cartesian coordinates x_i. The weights c_t and c_j are coupled coefficients depending on the trade-off between the time limit and desired smoothness of the robot movement. The constraints for pushing are first-order robot and object-environment dynamic constraints together with initial conditions. In cluttered environments an additional constraint is given by $\min_\gamma d(E\left[\mathbf{P}_O(t+\delta t)\right], p_d(\gamma)) < \rho$ where $d(\cdot)$ is the perpendicular distance of an object to the desired collision-free path $p_d(\gamma)$, $E[\cdot]$ is the expected value over object movements, and ρ is the tolerance for path tracking.

IV. Conclusion

Exploiting object manipulability in complex scenarios enables new capabilities and increases adaptiveness of a robot. A robot has to recognize specific object behaviours entailing strategies and the adaptation of pushing control. At the same time, movement primitives should be adapted using available sensor information. Such an adaptive control approach for push-manipulation is presented in this paper.

References

[1] Stefan Schaal, Jan Peters, Jun Nakanishi, and Auke Ijspeert. Learning movement primitives. In *International Symposium on Robotics Research (ISRR2003)*. Springer, 2004.

[2] Kevin M. Lynch and Matthew T. Mason. Stable pushing: Mechanics, controllability, and planning. *The International Journal of Robotics Research*, 15(6):533–556, 1996.

[3] Marcos Salganicoff, Giorgio Metta, Andrea Oddera, and Giulio Sandini. A vision-based learning method for pushing manipulation. In *In AAAI Fall Symposium Series: Machine Learning in Vision: What Why and*, 1993.

[4] D. Katz and O. Brock. Manipulating articulated objects with interactive perception. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 272–277, May 2008.

[5] F. Ruiz-Ugalde, G. Cheng, and M. Beetz. Fast adaptation for effect-aware pushing. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 614–621, Oct 2011.

[6] T. Hermans, J.M. Rehg, and A.F. Bobick. Decoupling behavior, perception, and control for autonomous learning of affordances. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 4989–4996, May 2013.

[7] M. Kopicki, S. Zurek, R. Stolkin, T. Morwald, and J. Wyatt. Learning to predict how rigid objects behave under simple manipulation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 5722–5729, May 2011.