Experimental Validation of Reference Spreading for Robotic Manipulation of Unmodeled Objects

Gijs van den Brandt

***Abstract*— This electronic document is a "live" template. The various components of your paper [title, text, heads, etc.] are already defined on the style sheet, as illustrated by the portions given in this document.**

I. INTRODUCTION

Automation has historically played a crucial role in the logistics industry. Our current way of living depends on autonomous systems for global transportation and warehousing. The growing labor shortage and increasing demand for online retail motivate further developments in the logistics sector [1].

An example of automation in logistics can be found in depalletizing, where object are removed from a pallet. Three different depalletizing approaches can be found in industry.

A logistical aspect where machines struggle to fully compete with humans is object manipulation. A practical examples of this is depalletizing. While robots are strong and consistent when manipulating objects, humans are versatile and swift. Robots are held back from faster performance because they must often slow down prior to making contact; establishing contact at a high velocity – an event referred to as an impact – could cause damage to the robot or its environment. On the contrary, humans intrinsically exploit impacts in the form of grabbing, bouncing and hitting. (give better explanation of depalletizing: manual and automatic approach)

The field of impact-aware control aims to better equip robots for making contact at high velocities. These impacts are paired with large contact forces that could damage the system. Previous work describes control using the maximum allowable impact velocity that complies with safety constraints such as limits for the contact force [2, 3]. This was combined with a compliant cover for the robot that reduces contact forces at impact, facilitating higher feasible impact velocities. Rather than using a soft cover, compliancy may also be achieved by designing a robot with low inertia and high backdrivability as was done in [4].

In addition to the large contact forces, a subject of interest is the velocity jump at the time of impact. Time misalignments between velocity jumps in the reference and in the actual system cause the velocity tracking error to peak[5], as is shown in Figure xxx. This error peak results in undesired control effort and should therefore be avoided. In [6], the robot's velocities are projected into an impact-invariant subspace based on the expected point of impact. As a result, impact-driven peaks in the velocity tracking error are reduced significantly. It is not always possible to describe a point of impact, however. Often times, impacts occur between surfaces rather than just points. Furthermore, corners of the surface may impact at diverging intervals in uncertain order during what is called near-simultaneous impacts, shown in Figure xxx.

The impact-aware control scheme called Reference Spreading [7] addresses error peaking caused by misaligned impacts. It operates on the basis of a tracking error that switches once an impact is detected. This concept is best explained at the hand of Figure xxx. The reference is split at the nominal impact time into an ante- and post impact reference. These references are then extended. Initially, the tracking error is based on the extended ante-impact reference, but this is switched to the post-impact reference once an impact is detected. Evidently, this can reduce the error peaking.

Reference spreading can also handle simultaneous impacts. [8] (explanation)

By addressing the peaking error, reference spreading facilitates faster object manipulation, making it interesting to industry if its effectivity can be proven in practice. Reference spreading for object manipulation has already been validated in simulations [8, 9]. Experimental validations have been limited to interaction with a fixed environment, however [10, 11]. The goal of this work is therefore to **provide a real-world implementation of reference spreading for practical object manipulation tasks.** To translate the results from simulation to reality, the following contributions are made:

1. Motion planning for impacts without object models: Generating a reference with velocity jumps that is coherent with the systems dynamics is challenging. One approach maps the ante-impact velocity to the post-impact velocity based on conservation of momentum [ref for impact map]. This requires a model of the environment, which is feasible in simulations with simplified dynamics, but challenging in reality. Impact-driven velocity jumps could instead be inferred experimentally. In previous studies [12], the control gains are reduced to zero upon detection of the impact while inferring an impact map, so that the velocity jump would not result in excessive motor torques. A different model-free motion planning strategy is proposed, which not only produces velocity-reference jumps that are coherent with the system dynamics, but also leverages human intuition to generate fluid motions before and after the impact. This is achieved by introducing a human in the loop by means of teleoperation.

2. Impact detection: The reference spreading scheme should switch between ante-, intermediate-, and post-impact references at the appropriate time. This requires an impact detection algorithm. Approaches in literature look either at position data [13] or external force estimations [11, 14, 15] for signs that could be caused by impacts. We show that these signs are necessary, but not sufficient conditions for an impact – only looking at position or contact force can result in false positives. To limit false detection of impacts, a novel impact detector that looks at both force and position data is proposed and evaluated.

3. Custom end effector:

(4. Intermediate impact phase controller:)

II. System overview

Considering the goal of evaluating reference spreading in a practical usecase, this work will focus on a dual-arm robotic setup. Having two arms increases the maximum payload. Furthermore, some object manipulation tasks, such as grabbing, require contact from multiple sides. The impact tasks that are considered in this work are stamping, swiping, grabbing, and tilting.

The Franka Emika robot [16] is used for the setup as it is affordable and, more importantly, capable of torque control which is critical for impact-aware manipulation. The robot uses harmonic drives that inherently have poor backdriveability, however, torque control is still possible thanks to the torque sensors. Franka Emika provides robot safety limits, such as maximum joint torques or velocities, that should not be exceeded, as this will trigger the

The system is limited in certain aspects to make it more representative of affordable setups in industry; in contrast to other object-manipulation works, we do not employ object pose estimation, environment models, or exteroceptive force/torque sensors.

(image of robots)

A. Robot dynamics

The robot dynamics can be described as

$$M(q)\ddot{q} + h(q,\dot{q}) = \tau_{cmd} + \sum_{i=1}^{n} J_i^T f_{c,i} \tag{1}$$

with inertia matrix M, joint accelerations $\ddot{q}$, centrifugal, coriolis and gravity terms h, and commanded torque τ_{cmd}. The robot has n contacts, and for each contact i there is an external contact wrench $f_{c,i}$. The wrench's contribution on joint level is related through contact jacobian J_i for which it holds that $\begin{bmatrix}\omega & \dot{p}\end{bmatrix}_i^T = J_i\dot{q}$, with ω the angular velocities of the contact body and $\dot{p}$ the Cartesian velocities of the contact point.

For modeling purposes, it is assumed that contact forces only act on the end effector body. These forces are modeled as a single contact wrench f_c acting on control point p, which is chosen as the intersection of robot link 5 and 7. This assumption allows for f_c to be estimated using methods described in section xxx. Furthermore, omitting dependency on q and $\dot{q}$ from for brevity results in

$$M\ddot{q} + h = \tau_{cmd} + J^T f_c. \tag{2}$$

III. Soft end effector design

A. Base controller

For articulated robot arms, it is convenient to control the end effector in Cartesian space, rather than directly controlling the robot's joints. In the case of the 7-DoF Franke Emika robot, this leaves one redundant DoF however. Furthermore, despite the desire to control the end effector Carthesian space,

This can be accomplished with an impedance controller.

The desired impedance behavior is

$$\Lambda \begin{bmatrix} d_{\dot{\omega}} - \dot{\omega} \\ d_{\ddot{p}} - \ddot{p} \end{bmatrix} + D \begin{bmatrix} d_{\omega} - \omega \\ d_{\dot{p}} - \dot{p} \end{bmatrix} + K \begin{bmatrix} e_{rot} \\ d_p - p \end{bmatrix} = d_{f_c} - f_c : \tag{3}$$

a mass spring damper with with Cartesian-space inertia, damping and stiffness matrices Λ, D and K, and rotation tracking error e_{rot} as defined in Appendix xxx. Desired values are denotes as $d_{(\cdot)}$, e.g., the desired value for $\ddot{p}$ is $d_{\ddot{p}}$.

Stiffness K is typically chosen as a diagonal matrix. The damping matrix is determined following $D = 2(\Lambda K)^{\frac{1}{2}}$ which guarantees stable behavior when K and Λ are symmetric. Furthermore, For the inertia matrix, two options were considered. Choosing a diagonal matrix Λ decouples the accelerations w.r.t. to the position error, resulting in better tracking in free motion. This approach was used in [17]. In this work, however, the task-space inertia is set to match the joint-space inertia following $\Lambda^{-1} = JM^{-1}J^T$, decoupling the contact force w.r.t. the position error for better performance during contact. Further motivation for this decision is provided in Appendix xxx.

Based on 3 we can determine target task-space accelerations, $t_{\dot{\omega}}$ and $t_{\ddot{p}}$, following

$$\begin{bmatrix} t_{\dot{\omega}} \\ t_{\ddot{p}} \end{bmatrix} = \begin{bmatrix} \dot{\omega} \\ \ddot{p} \end{bmatrix} + \Lambda^{-1} f_c \tag{4}$$

$$= \begin{bmatrix} d_{\dot{\omega}} \\ d_{\ddot{p}} \end{bmatrix} + \Lambda^{-1} \left(D \begin{bmatrix} d_{\omega} - \omega \\ d_{\dot{p}} - \dot{p} \end{bmatrix} + K \begin{bmatrix} e_{rot} \\ d_p - p \end{bmatrix} - d_{f_c} \right). \tag{5}$$

Note the exclusion of f_c as the exerted contact wrench is not modeled and therefore unknown. Quadratic Programming is used to find τ_{cmd} so that the weighted squared error between the desired and actual accelerations is minimized while accounting for the system's dynamics (x) and safety constraints (x).

The described impedance task only tracks 6 DoF's however, leaving one redundant DoF of the Franka Emika robot. To resolve this redundancy, one more task which describes the target acceleration of the first robot joint is added. This so-called posture task with stiffness k_p is given by

$$\ddot{q}_{1,t} = 2\sqrt{k_p}\dot{q}_1 + k_p(q_1 - q_{1,d}). \tag{6}$$

An overview of the used control parameters is given in Table xxx.

Parameter	Value
$q_{1,d}$	0 rad
k_1	50 Nm/rad
k_2	800 N/m
k_p	50 Nm/rad
K	$\begin{bmatrix} k_1 I & 0 \\ 0 & k_2 I \end{bmatrix}$

IV. Reference spreading controller

Reference spreading reduces control effort by cleverly redefining the tracking target. These target definitions differ between the ante, intermediate, and post-impact phase based on separate ante- and post impact references. Impact detection is used to switch between the impact phases at the appropriate time. This section describes the three key components of reference spreading: impact detection, reference formulation, and tracking error definitions.

A. Impact detection

B. Reference formulation

(subscript r indicates a reference, superscripts a and p stand for ante-impact and post-impact phase respectively)

C. Target definition

Table IV-C shows an overview of the target definitions during the three impact phases. These targets are used for the impedance controller. During the ante- and post impact phase, the target is equal to the respective ante- or post impact reference. For the intermediate phase, multiple options are considered.

Intermediate phase option 0 is equivelant to the post-impact mode, meaning that the controller effectively jumps from ante- to post impact mode directly.

In [8], it is mentioned that following the post-impact reference does not make sense during the intermediate mode where contact is not yet completed. Instead, the ante-impact reference should be targeted until contact completion, with exception of the reference velocity which is causing the error peak. Setting the target velocity to the current velocity, i.e., $\dot{p}_t = \dot{p}$, causes the velocity tracking error to be zero. This aligns with intermediate option 1.

Option 2 was inspired by [11] where the velocity target was set to zero following from the observation that damping is beneficial for reducing oscilations.

Upon the transition from intermediate to the post-impact phase, intermediate option 1 and 2 will experience a jump in the targets p_t and F_t as the respective ante and post impact references are not guaranteed to coincide. To address this issue, we propose mixing of the ante and post impact reference during the intermediate phase in option 3. Mixing value γ equals zero at the start of the intermediate mode, and increments linearly up to 1 at the end of the intermediate phase. Furthermore, opposed to option 2 where the damping target is a velocity of zero, option 3 employs damping with respect to a target velocity equal to the post impact reference.

TABLE I

Impedance target definition

Definition of the targets p_t, $\dot{p}_t$, and F_t during the different impact phases. Multiple options for the intermediate phase are considered.

			Intermediate			
	Ante	Post	0	1	2	3
p_t	p_r^a	p_r^p	p_r^p	p_r^a	p_r^a	$\gamma p_r^a + (1-\gamma) p_r^p$
$\dot{p}_t$	$\dot{p}_r^a$	$\dot{p}_r^p$	$\dot{p}_r^p$	$\dot{p}$	0	$\gamma p + (1-\gamma) \dot{p}_r^p$
F_t	F_r^a	F_r^p	F_r^p	F_r^a	F_r^a	$\gamma F_r^a + (1-\gamma) F_r^p$

References

[1] A. Dekhne, G. Hastings, J. Murnane, and F. Neuhaus, “Automation in logistics: Big opportunity, bigger uncertainty,” *McKinsey Q*, pp. 1–12, 2019.

[2] N. Dehio and A. Kheddar, “Robot-Safe Impacts with Soft Contacts Based on Learned Deformations,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1357–1363, Xi’an, China: IEEE, May 30, 2021.

[3] N. Dehio, Y. Wang, and A. Kheddar, “Dual-Arm Box Grabbing With Impact-Aware MPC Utilizing Soft Deformable End-Effector Pads,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5647–5654, Apr. 2022.

[4] H. Song, Y.-S. Kim, J. Yoon, S.-H. Yun, J. Seo, and Y.-J. Kim, “Development of Low-Inertia High-Stiffness Manipulator LIMS2 for High-Speed Manipulation of Foldable Objects,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4145–4151, Madrid: IEEE, Oct. 2018.

[5] J. Biemond, N. van de Wouw, W. Heemels, R. Sanfelice, and H. Nijmeijer, “Tracking control of mechanical systems with a unilateral position constraint inducing dissipative impacts,” in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 4223–4228, Maui, HI, USA: IEEE, Dec. 2012.

[6] W. Yang and M. Posa, “Impact Invariant Control with Applications to Bipedal Locomotion,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5151–5158, Prague, Czech Republic: IEEE, Sep. 27, 2021.

[7] A. Saccon, N. van de Wouw, and H. Nijmeijer, “Sensitivity analysis of hybrid systems with state jumps with application to trajectory tracking,” in *53rd IEEE Conference on Decision and Control*, pp. 3065–3070, 2014.

[8] J. J. van Steen, N. van de Wouw, and A. Saccon, *Robot control for simultaneous impact tasks via QP based reference spreading*, Nov. 9, 2021.

[9] S. Zwart, “Impact-Aware Learning from Demonstration,” Master’s thesis, Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, 2019.

[10] M. Rijnen, A. Saccon, and H. Nijmeijer, “Reference Spreading: Tracking Performance for Impact

Trajectories of a 1DoF Setup," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 3, pp. 1124–1131, 2020.

[11] S. S. Uitendaal, "Teaching Robots Interaction Task by Performing Demonstrations," Master's thesis, Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, 2022.

[12] I. Aouaj, V. Padois, and A. Saccon, "Predicting the Post-Impact Velocity of a Robotic Arm via Rigid Multibody Models: An Experimental Study," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[13] M. Rijnen, A. Saccon, and H. Nijmeijer, "Motion Signals With Velocity Jumps: Velocity Estimation Employing Only Quantized Position Data," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1498–1505, Jul. 2018.

[14] B. Proper, "Aim-Aware Collision Monitoring for Robot-Environment Edge Impacts," Master's thesis, DC 2021.081, Department of Mechanical Engineering, Eindhoven University of Technology, 2021.

[15] B. Proper, "Validation of Numerical Simultaneous Impact Classification Algorithm," Internship, Department of Mechanical Engineering, Eindhoven University of Technology, 2022.

[16] S. Haddadin *et al.*, "The Franka Emika Robot: A Reference Platform for Robotics Research and Education," *IEEE Robotics & Automation Magazine*, vol. 29, no. 2, pp. 46–64, Jun. 2022.

[17] S. B. van Oorschot, "Design and numerical validation of a dual-arm telerobotic simulation for object grabbing," Master's thesis, Department of Mechanical Engineering, Eindhoven University of Technology, 2022.