Model-based Cartesian Impedance Control

Real Robot Challenge Phase III Report

Anonymous Author(s)[1]

[1]Affiliation
Email: Email

Abstract

For this last phase, we adapted our model-based cartesian impedance controller in three ways: An improved pre-grasping position controller, refined grasping points and obtaining new hyperparameters through Bayesian optimization on the real system. Combined, these three adaptations enabled us to tackle the significantly more challenging cuboid object, which required more delicate manipulation compared to the cube. Videos can be found at: `sites.google.com/view/robotchallenge-modelbased`

1 Introduction

In Phases I & II, we proposed a model-based Cartesian impedance controller (CIC) [1, 2, 3, 4] of the TriFinger robotic platform [5] for the Real Robot Challenge (RRC). The motivation for such a structured approach is balance between inductive biases and adaptivity. Encoding prior knowledge like the dynamics model and goal state avoids excessive learning, while the adaptivity of the impedance structure should allow for sim-to-real transfer without explicit consideration. While purely data-driven robot learning is a noble pursuit, the time and resource limitations of the RRC encourage maximal usage of prior knowledge.

In Phase I, the controller had several limitations. For one, each finger was controlled independently. This severely limited the ability to perform orientation control, which is best achieved by moments generated by several fingers acting in a coordinated fashion.

Therefore, in Phase II, we proposed to overcome these limitations by adding an holistic formulation that considers all three fingers, a task decomposition for complex manipulation sequences and added Bayesian optimization of the hyperparameters.

The Phase II controller had two key weaknesses: The IK-based pre-grasp manoeuvre had no timing guarantees, which was the source of random catastrophic failures. The grasp quality was also a source of fragility, as we were limited to form-closure rather than force-closure.

While the Phase II additions improved the performance, the existing issues and control strategy had to be adjusted for the task of manipulating the fine cuboid, which is smaller and lighter than the previous cube. The biggest challenge was a surprise, as it actually pertained to the pre-grasp movements. Previously, we controlled the robot solely using torque commands. However, when approaching the object, iteratively solving the inverse kinematics (IK) could result in timing issues that lead to undesired movements of the fingers and disturbing the object. Whereas the heavier cube of Phase II was robust against these movements, in this phase, these disturbances significantly displaced the cuboid, such that it was often difficult to recover. This phenomena was also absent in the simulation, so it required tuning on the real system. Thus, we switched to position control for the pre-grasp movements and determined additional via-points that should stabilize the manoeuvre. Moreover, due to the change in the object, we had to adapt the grasping points and re-run the Bayesian optimization (BO) [6] of the hyperparameters on the real system.

Videos illustrating the performance of the presented approach are provided [1].

The report is structured as follows: Section 2 recaps the controller formulation, Section 3 details the finger placement and pre-grasp movements, and Section 4 describes the use of Bayesian optimization. Section 5 provides the team's conclusion.

[1] https://sites.google.com/view/robotchallenge-modelbased

2 Cartesian Model-based Impedance Control

In this section, we summarize the controller and explain the holistic formulation.

Grasping and Position Control For the desired cartesian position of the ith fingertip $\boldsymbol{X}_i$, we define $\bar{\boldsymbol{X}}_i$ to be the error between this tip position and the cuboid's centre of mass $\boldsymbol{X}_c$, so $\bar{\boldsymbol{X}}_i = \boldsymbol{X}_c - \boldsymbol{X}_i$. We then define an impedance controller for $\bar{\boldsymbol{X}}_i$, a second order ODE that can be easily interpreted as a mass spring damper system with parameters $\{\boldsymbol{M}, \boldsymbol{D}, \boldsymbol{K}\}$,

$$\boldsymbol{M}\ddot{\bar{\boldsymbol{X}}}_i + \boldsymbol{D}\dot{\bar{\boldsymbol{X}}}_i + \boldsymbol{K}\bar{\boldsymbol{X}}_i = \boldsymbol{f}_i. \tag{1}$$

For the task of grasping the cuboid, each finger is controlled independently. Since the cuboid's centre of mass is estimated using a vision-based system, the damping factor was zeroed for fast control. Converting this cartesian space control law back to joint coordinates results in $\boldsymbol{\tau}_{1,i} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}^{-1}\ddot{\bar{\boldsymbol{X}}}$, where $\boldsymbol{\tau}_{1,i}$ denotes the torques to be applied to finger i, $\boldsymbol{q}$ the joint configuration and $\boldsymbol{J}$ the Jacobian. The natural adaptivity of this impedance control law ensures a stable grasp of the cuboid.

To perform cuboid pose control, we follow the ideas presented in previous CIC literature [4] and design a proportional control law that perturbs the cuboid's centre based on the goal position $\boldsymbol{X}_g$, $\hat{\boldsymbol{X}}_c = \boldsymbol{X}_c + K_1(\boldsymbol{X}_g - \boldsymbol{X}_c)$. Replacing $\boldsymbol{X}_c$ by $\hat{\boldsymbol{X}}_c$ in the equation above results in our impedance-based grasping and position control law. It ensures that the cuboid is grabbed and held (mostly by choosing the stiffness $\boldsymbol{K}$) and moves the cuboid to the desired goal location. K_1 determines the evolution of the reference position and allows to control how fast the cuboid moves to its target.

Force-based, Holistic Components Despite the functionality of the initial controller, it did not consider the fingers as a whole, and so was limited in controlling orientation of the cuboid. Contact forces were also passively applied rather than explicitly considered. To incorporate these additional considerations, we superimpose four torques. First is the already introduced position control and gravity compensation, which is added with three contact and rotational terms explained in the following, such that $\boldsymbol{\tau}_i = \sum_{j=1}^{4} \boldsymbol{\tau}_{j,i}$.

To also allow directly specifying the force applied by each finger, we introduce an additional component $\boldsymbol{\tau}_{2,i} = \boldsymbol{J}^\intercal \boldsymbol{F}_{2,i}$, where $\boldsymbol{F}_{2,i}$ is the force applied by finger i. We chose $\boldsymbol{F}_{2,i}$ to be in the direction of the surface normal of the face where finger i touches the cuboid ($\boldsymbol{F}_{2,i} = K_2\boldsymbol{d}_i$). However, to not counteract the impedance controller, the resulting force of this component $\boldsymbol{F}_{\text{res}} = \sum_i \boldsymbol{F}_{2,i}$ should account to zero. We ensure this by solving

$$-\boldsymbol{F}_{\text{res}} = [\boldsymbol{J}^{-\intercal}, \boldsymbol{J}^{-\intercal}, \boldsymbol{J}^{-\intercal}][\boldsymbol{\tau}_{3,1}, \boldsymbol{\tau}_{3,2}, \boldsymbol{\tau}_{3,3}]^\intercal, \tag{2}$$

for $\boldsymbol{\tau}_{3,i}$. All previous components ensure a stable grasp closure. This is essential for the following orientation control law. Neglecting the cuboid's exact shape, we model the moment that is exerted onto the cuboid as $\boldsymbol{\Omega} = \sum \boldsymbol{r}_i \times \boldsymbol{F}_{4,i} = \sum \boldsymbol{S}_{r_i}\boldsymbol{F}_{4,i}$, where $\boldsymbol{r}_i = -\bar{\boldsymbol{X}}_i/|\bar{\boldsymbol{X}}_i|_2$ denotes the vector pointing from the cuboid's centre towards the finger position, $\boldsymbol{S}_{r_i}$ the respective skew-symmetric matrix, and $\boldsymbol{F}_{4,i}$ an additional force that should lead to the desired rotation. The goal is now to realize a moment proportional to the current rotation errors, which are provided in the form of an axis of rotation $\boldsymbol{r}_{\boldsymbol{\phi}}$ and its magnitude ϕ. Thus, the control law yields $\boldsymbol{\Omega} = K_3\phi\boldsymbol{r}_{\boldsymbol{\phi}}$. We achieve $\boldsymbol{\Omega}$ by solving

$$\boldsymbol{\Omega} = [\boldsymbol{S}_{r_1}\boldsymbol{J}^{-\intercal}, \boldsymbol{S}_{r_2}\boldsymbol{J}^{-\intercal}, \boldsymbol{S}_{r_3}\boldsymbol{J}^{-\intercal}][\boldsymbol{\tau}_{4,1}, \boldsymbol{\tau}_{4,2}, \boldsymbol{\tau}_{4,3}]^\intercal \tag{3}$$

for $\boldsymbol{\tau}_{4,i}$. Associated parameters are tuned using Bayesian optimization.

3 Finger Placement and Pre-Grasp Movement

We further combine the previously introduced control law with a simple finger placement heuristic. Compared to the previous phase, we ended up only using two fingers to control the cuboid. We place the fingers such that the two fingers occupy the centres of the two long faces that are perpendicular to the ground plane. The assignment is computed based on the current distance of the fingers to the desired positions. Initially, we also planned to place the third finger on one of the small faces to enable fine orientation control, especially once the cuboid is lifted.

However, we failed to come up with a robust implementation of this strategy in time and found better results using the two fingers only. The main case of failure was that the force applied by the third finger resulted in one of the other fingers losing contact and therefore dropping the cuboid. We hypothesis that force-closure, rather than form-closure, is required to make this delicate three finger grasp work reliably.

As the cartesian impedance controller does not guarantee the grasping points explicitly, it is essential to add a pre-grasping manoeuvre. We drive the fingers to locations such that activating the impedance control law results in the desired contact points. Therefore, we define a pre-grasping trajectory consisting of three via-points as illustrated in Figure 1. In the previous phases, we used torque control to drive the fingers to the desired locations. However, as this sometimes resulted in violating the timing and poor control, we changed to position control. After reaching the third via-point, we switch back to the torque-based impedance control law which does not contain any iterative solving of the IK, and reliably runs at the specified frequency. Replacing the via-points with a smoother motion plan is a planned future improvement. Improving the grasp point heuristic is another open area for improvements. We believe the kinematics of the system is not flexible enough that simple finger placement and impedance motion guarantees grasp maintenance. Therefore, it is likely that this needs to be incorporated into the grasp choice and motion.

4 Bayesian Optimization of Hyperparameters

From the formulation detailed in Section 2, the CIC mainly depends on six hyperparameters $(\boldsymbol{K}, K_1, K_2, K_3)$. The overall control strategy contains more parameters, such as the via-points for approaching the desired grasp locations or thresholds that define when to switch to the next 'primitive' controller, i.e. when to switch from moving the cuboid in the ground plane to lifting it. Bayesian optimization was used to perform sample-efficient black-box optimization, using the BoTorch library [7]. Optimizing the hyperparameters can also be viewed as correcting for modelling error through parameter tuning. In

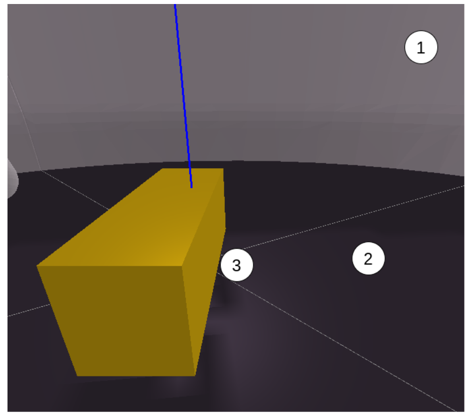

Figure 1: Illustration of the three via-points of the pre-grasp manoeuvre. The first point is chosen such that it can be reached by any finger without colliding with the cuboid, while points 2 and 3 should allow a smooth approaching movement.

the last phase, we found the importance of tuning the contact-based hyperparameters (impedance stiffness and the force scalar). Having different stiffnesses for the xy and z direction was found to dramatically improve performance. For successfully manipulating the smaller object, optimizing the parameters of the pre-grasp manoeuvre had a big impact. In particular, we used BO to tune the location of third via-point since it heavily influenced the grasp success.

When applying BO on the real system, we only optimized the key contact-based and pre-grasp parameters to reduce the dimensionality of the search space. To counteract the stochasticity of the real platform and to ensure generalization of the identified parameters, we performed several rollouts each iteration.

5 Conclusion and Outlook

For this phase, our time was invested in robustifying the pre-grasp motion, rather than improving the core manipulation algorithm, due to the change in object. While improvements were made, we believe more work needs to be done. In particular, the grasp robustness needs to be improved. Further, we believe force-closure, combined with motion planning that can maintain this grasp, is required.

References

[1] Kevin M. Lynch and Frank C. Park. *Modern Robotics: Mechanics, Planning, and Control.* 2017.

[2] Thomas Wimböck, Christian Ott, Alin Albu-Schäffer, and Gerd Hirzinger. Comparison of object-level grasp controllers for dynamic dexterous manipulation. *The International Journal of Robotics Research*, 2012.

[3] Luigi Biagiotti, Hong Liu, Gerd Hirzinger, and Claudio Melchiorri. Cartesian impedance control for dexterous manipulation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003.

[4] Martin Pfanne, Maxime Chalon, Freek Stulp, Helge Ritter, and Alin Albu-Schäffer. Object-level impedance control for dexterous in-hand manipulation. *IEEE Robotics and Automation Letters*, 2020.

[5] Manuel Wüthrich, Felix Widmaier, Felix Grimminger, Joel Akpo, Shruti Joshi, Vaibhav Agrawal, Bilal Hammoud, Majid Khadiv, Miroslav Bogdanovic, Vincent Berenz, Julian Viereck, Maximilien Naveau, Ludovic Righetti, Bernhard Schölkopf, and Stefan Bauer. Trifinger: An open-source robot for learning dexterity, 2020.

[6] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, 2012.

[7] Maximilian Balandat, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham, Andrew Gordon Wilson, and Eytan Bakshy. BoTorch: Programmable Bayesian Optimization in PyTorch. *arxiv e-prints*, 2019.