Dyadic collaborative Manipulation through Hybrid Trajectory Optimization

Theodoros Stouraitis
The University of Edinburgh & HRI-EU
theodoros.stouraitis@ed.ac.uk

Iordanis Chatzinikolaidis
The University of Edinburgh
i.chatzinikolaidis@ed.ac.uk

Michael Gienger
Honda Research Institute Europe (HRI-EU)
michael.gienger@honda-ri.de

Sethu Vijayakumar
The University of Edinburgh
sethu.vijayakumar@ed.ac.uk

Abstract: This work provides a principled formalism to address the joint planning problem in dyadic collaborative manipulation (DcM) scenarios by representing the human's intentions as task space forces and solving the joint problem holistically via model-based optimization. The proposed method is the first to empower robotic agents with the ability to plan in hybrid spaces – optimizing over discrete contact locations, continuous trajectory and force profiles, for co-manipulation tasks with varied dyadic objective goals. This ability is particularly important in large object manipulation scenarios that typically require change of grasp-holds. The task of finding the contact points, forces and the respective timing of grasp-hold changes are carried out by a joint optimization using non-linear solvers. We demonstrate the efficacy of the optimization method by investigating the effect of robot policy changes (trajectories, timings, grasp-holds) based on changes in collaborative partner policies using physically based dynamic simulations. We also realize, in hardware, effective co-manipulation of a large object by the human and the robot, including eminent grasp changes as well as optimal dyadic interactions to realize the joint task.

Keywords: Optimal Control, physical Human-Robot Collaboration, Manipulation, Trajectory Optimization

1 Introduction

With Dyadic collaborative Manipulation (DcM) we refer to a set of two individuals jointly manipulating an object, as shown in Fig. 1. The two individuals partner together to form a distributed system, augmenting their manipulation abilities. Such individuals can be either humans or robots. In scenarios where both individuals are humans, the collaboration is natural as we humans are adept at co-manipulation. Nevertheless, our understanding of the mechanisms of joint action and the development of robot partners are still subject of research [1]. Endowing robotic agents with collaborative capabilities is of crucial importance towards the development of robotic partners.

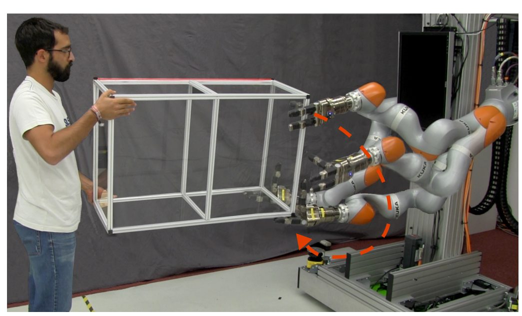


Figure 1: DcM scenario during change of contact.

Early work by Sheridan in [2], identified eight core challenges of human-robot communication, with two of them being: i) the need to acquaint both humans and robots with models of their partners, and ii) the need to regulate the interaction of distributed decision-making systems, typically referred as *mixed initiative* systems. Ajoudani et al. [3] summarized the strategies used to equip robots with interaction capabilities and pinpointed that research on human-robot interaction models is still at its infancy. Thus, in this work we focus on how a robot policy can be partner-informed and flexible towards complying with the requirements of DcM scenarios.

2nd Conference on Robot Learning (CoRL 2018), Zürich, Switzerland.

Accounting for partner's actions is central in Human-Robot-Collaboration (HRC) domain and there are four prevailing schools of thought.

Control focused: Agravante et al. [4] used an impedance port to accommodate partner's actions and demonstrate collaboratively carrying of a table, while balancing a free-rolling ball on top of it. In [5] the authors presented a load sharing framework based on force space redundancy. In both cases, the partner is treated as an external disturbance to the system.

Estimation focused: Thobbi et al. [6] used a confidence measure of the human's goal prediction to alter between reactive and proactive robot behaviour. Huang et al. [7] utilized this idea and evaluated adaptive, solely reactive and solely proactive robot behaviours towards improving the dyad's performance and the user experience. These methods rely on the accurate prediction of the human's intentions to switch between discrete robot behaviours.

Interaction focused: Maeda et al. [8] and Rozo Castañeda et al. [9] proposed methods to transfer adaptive hand-overs and variable impedance behaviours to robots from kinesthetic demonstrations. A data-driven method to extract the interaction constraints in hand-over tasks was proposed by Vogt et al. [10]. The constraints were then used to form online robot responses. These methods couple together the policies of the agent, the partner, and the progress of the task, to learn a direct mapping which is used to generate online adaptive robot responses.

Partner model focused: Maeda et al. [11] developed a polynomial based online human motion target prediction to continuously update the goal of the robotic agent. Gribovskaya et al. [12] proposed a method that learns the task model and a feed-forward controller to comply with it. Such methods are elegant, as each entity of the interaction is modelled separately. Our work falls under this school's area as we believe that separate models for the partner and the task are important towards obtaining appropriately reasoned and generalizable robot behaviours.

In addition to the human action recognition and prediction, an other central aspect in HRC domain is: the attributes of the robot motion being regulated to fulfill the task in response to human's actions and intentions.

In [13] attributes of the task space, such as the object's trajectory, are optimized to facilitate human ergonomics. In [14, 15] phase-based adaptation during co-manipulation is realized through exploitation of synchronization patterns in the time domain, resulting in turn-taking collaboration. Further, a large variety of methods has focused on utilizing the dynamic properties of the interaction, like interaction forces. Inverse dynamics approaches in [16, 17] concentrate on the torque and force regularization, while others [12, 18] adapt the impedance characteristics of the robot online to accommodate for partner's actions. However, a central aspect of manipulation is the selection of the appropriate contact locations on the object [19]. Accordingly, the exploitation of the contact space between the manipulated object and the individuals in DcM scenarios is vital. To the best of our knowledge, the topic of contact adaptation within co-manipulation scenarios has not been addressed yet. Thus, the focal point of this work is how a robot policy can utilize the redundancy in the contact space, towards addressing human-robot requirements in DcM scenarios.

The contributions of this paper are:

Dyadic planning formalism: We formalize joint planning in DcM tasks as the problem of finding the appropriate actions to control the manipulated object, given an estimate of the partner's policy. The formalization provides a principled method for dyadic joint action. The method is able to generate different instances of the policy either due to changes of the partner's policy or due to changes of the dyadic setup.

Hybrid optimal control for manipulation: We present a holistic model-based optimization method, based on trajectory optimization (TO) framework that allows robotic agents to simultaneously exploit the redundancy in the agent's i) forces, ii) contact locations, iii) actions timings, and iv) the object's trajectory, to obtain an optimal solution in manipulation tasks.

Overall, our work is a first step towards exploiting hybrid action spaces in order to meet the joint planning requirements in DcM scenarios. This paper is organized as follows: Section 2 presents the formulation of the addressed problem. The details on the TO method used are given in Section 3. Section 4 presents the evaluation of the method and the experimental results. Finally, Section 5 points out the conclusions of this study and discusses promising research directions.

2 Dyadic planning

Fig. 2 provides a graphical representation of DcM as a system. It is separated into two components, the interaction attributes of the dyadic setup and the action generation of an individual. The binding between the two partners is both in physical and in mental terms. The physical coupling arises due to the object, while the intentions of the individuals are naturally correlated due to the common objectives of the dyad.

Nomenclature: The notation used in this paper follows:

Physical quantities:

- $\boldsymbol{\lambda} \in \mathbb{R}^{\nu}$: Net forces applied by the partner
- $\mathbf{f}^i \in \mathbb{R}^{\nu}$: Forces applied by agent's i_{th} limb
- $\mathbf{c}^i \in \mathbb{R}^{\nu}$: Agent's i_{th} end-effector position
- $\mathbf{K}^a, \mathbf{D}^a \in \mathbb{R}^{\nu}$: Agent's stiffness and damping
- $\mathbf{q} \in \mathbb{R}^{n}$: Agent's configuration
- $\mathbf{y}_{t:T} \in \mathbb{R}^{\nu \times K}$: Pose trajectory of the object
- $\Delta T \in \mathbb{R}$: Agent's actions timings
- $\nu \in \mathbb{R}$: Dimensionality of the task
- $n \in \mathbb{R}$: Dimensionality of the robot c-space
- $K \in \mathbb{R}$: Number of knots
- $\iota \in \mathbb{R}$: Number of limbs

Non-physical quantities:

- $\pi_a, \pi_p \in \Xi$: Agent's and partner's policy
- Ξ : Function space of all possible trajectories
- S_d : Dyadic setup
- M_{task} : Description of the manipulation task

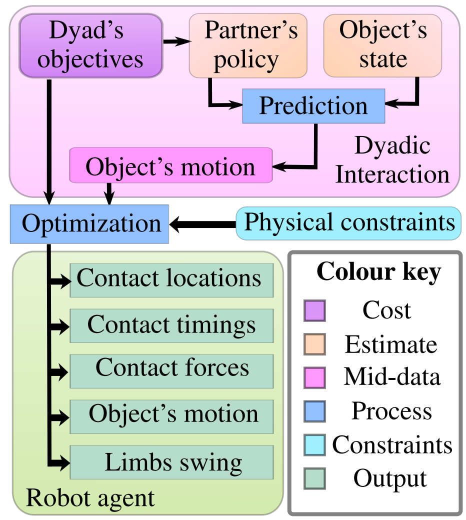


Figure 2: Modular description of DcM as a system.

Formulation: In the introduction of the paper we motivated the need of agent's capabilities within DcM tasks and we emphasized the lack of methods that are able to produce policies which can exploit hybrid spaces, like the force-contact space. Formally, the full control policy of an agent participating in DcM tasks is defined as in Eq. (1),

$$\mathbf{f}^i, \mathbf{c}^i, \mathbf{K}^a, \mathbf{D}^a, \mathbf{q} = \pi_a \quad (1) \qquad\qquad \mathbf{f}^i, \mathbf{c}^i = \pi_a \quad (2)$$

however as this work focuses on the generation of hybrid plans, the policy of the agent can be reduced to the form shown in Eq. (2). Additionally, the policies of the two individuals are related forming the dyadic interaction.

$$\pi_a = \mathcal{f}_{\pi}(\hat{\pi}_p, S_d, M_{task}) \quad (3) \qquad\qquad \mathbf{y}_{t:T} = \mathcal{f}(\mathbf{f}^i, \mathbf{c}^i, \pi_p) \quad (4)$$

We represent this relationship with Eq. (3), which indicates the dependency of the agent's policy to the estimated policy of the partner $\hat{\pi}_p$, the dyadic setup S_d, and the manipulation task M_{task}. The actual task is the object's motion, which is governed by the blend of the two individuals' policies. Hence, the trajectory of an object can be described as a function of the two policies, as shown in Eq. (4). The aim in DcM scenarios is to obtain the agent's optimal policy π_a^* that depends on the dyadic setup, the manipulation task and the partner's estimated policy.

$$\begin{aligned} \pi_a^* = \underset{\mathbf{f}^i, \mathbf{c}^i, \mathbf{y}_{t:T}, \Delta T}{\arg\min} \quad & \int_0^T \mathcal{c}(S_d, M_{task}) dt \\ \text{s. t.} \quad & \mathcal{h}(\mathbf{y}_t, \hat{\pi}_p, M_{task}) = 0, \\ & \mathcal{g}(\hat{\pi}_p) \leq 0 \end{aligned} \tag{5}$$

With Eq. (5) we formalize dyadic planning; by introducing the idea of considering the partner's actions into the motion plans of the robot through the constraint functions $\mathcal{h}$ and $\mathcal{g}$, while the objectives of the dyad are met through the cost function $\mathcal{c}$.

3 DcM solution with Hybrid model-based optimization

Hybrid problems have been addressed with hierarchical structures [20], which typically provide sub-optimal solutions. Single agent hybrid manipulation has been demonstrated, also being robust to uncertainties with continuous [21] or discrete [22] actions. These works assume quasi-static environments, which limits their applicability on 2D tabletop examples only. On the other hand, there exists an important duality between manipulation and locomotion [23]. We draw inspiration from a family of methods presented in [24, 25], that use a variety of model-based optimal control formulations that are not restricted by quasi-static stability assumption, and have successfully addressed hybrid locomotion problems. In this family of methods two observations are of key importance: i) motions through contact have phases and ii) the contact set remains invariant within each phase.

Background on TO: TO methods address the problem of finding the optimal open-loop control policy [26]. Such a policy can be represented as a function that lives in the function space Ξ. The transformation process – from the space of functions to the space of parameters – is called transcription. The specific transcription method used in this work is called direct collocation, and allows us to describe the function of the policy with a set of knots points referred as mesh and the respective connection segments. These knots and their segments form a trajectory, shaped by the connectivity of neighboring knots, which is enforced with the *collocation* constraints.

3.1 Hybrid policy through trajectory optimization:

The policy of the agent π_a lives in a non-linear hybrid manifold, from which we sample trajectories utilizing the TO framework. To achieve this, each k_{th} knot of the trajectory is set as a vector of the following decision variables: i) the pose of the object $\mathbf{y}_k$, ii) action timings $\Delta\mathbf{T}_k$, iii) the contact locations $\mathbf{c}_k^i$, and iv) the contact forces $\mathbf{f}_k^i$. These are the quantities of interest found in Eq. (2), Eq. (4), and Eq. (5). Therefore, we can group the decision variables of the optimization method in a vector $\mathbf{s}$ that includes both states and actions as in Eq. (6). Using the vector $\mathbf{s}$, the problem of finding the optimal policy π_a^* can be transcribed to a non-linear constrainted optimization in the form of Eq. (5).

$$\mathbf{s} = [\mathbf{y}_k, \dot{\mathbf{y}}_k, \mathbf{c}_k^i, \dot{\mathbf{c}}_k^i, \mathbf{f}_k^i, \Delta\mathbf{T}_k]^T, \quad \forall k \in \{0, .., K\} \tag{6}$$

In contrast to [24], we use a structure-free representation of the decision variables, not to avoid restricting the resulting trajectories to any particular class of functions, e.g. polynomials. Next, we describe the phase-independent constraints applied to all the knots of the trajectory and the phase-specific constraints enforced on bundles of knots that belong in the same phase of the motion. Separating the motion in phases with different constraints allows us to explicitly address the complementary problem described below.

Phase-independent constraints: We list here the set of constraints applied at all the knots of the trajectory. The integration function, f_m, is implemented with a trapezoidal quadrature and ϕ_a defines the reachable area of the agent's end-effectors, referred as arms workspace.

- Dynamics of object's CoM:
$$[\mathbf{y}_{k+1} \quad \dot{\mathbf{y}}_{k+1}]^T = f_m(\mathbf{y}_k, \dot{\mathbf{y}}_k, \ddot{\mathbf{y}}_k, \boldsymbol{\lambda}, \mathbf{f}_k^i, \mathbf{c}_k^i, \Delta\mathbf{T}_k) \tag{7}$$
- Initial state of the object's CoM: $\mathbf{y}_0 = \mathbf{y}_0^*$ and $\dot{\mathbf{y}}_0 = \dot{\mathbf{y}}_0^*$ (8)
- Desired final state of the object's CoM: $\mathbf{y}_K = \mathbf{y}_K^*$ and $\dot{\mathbf{y}}_K = \dot{\mathbf{y}}_K^*$ (9)
- Kinematic limits of the agent's end-effectors (*Box constraints*): $\mathbf{c}_k^i \in \phi_a$ (10)
- Upper bound on the total time of the motion: $\sum_0^K \Delta\mathbf{T}_k \leq T_K$ (11)

The problem of intermittent contact boils down to a Complementary Problem (CP), defined as $\dot{\mathbf{c}}_k^T \mathbf{f}_k = 0$. The CP constraint exacerbates the convergence properties of the optimization problem as it introduces discontinuities in the dynamics. Posa et al. [27] proposed a relaxation of the CP constraint, to solve problems that require trajectories through contact. The concept of phases was introduced by Mordatch et al. [25], yet as all constraints were enforced as part of the cost function; physically inaccurate motions can be generated. In [24] the authors introduced a parameter that defines the exact number of contact changes, to allow efficient solving of the problem with hard constraints.

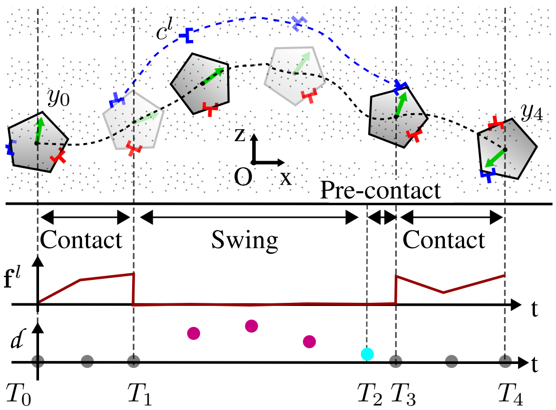


Figure 3: Hybrid motion plan with one grasp-hold change. The grey dotted area on top illustrates the physical space (x,z,θ). Orientation θ is illustrated with the green arrow in the object. The force $\mathbf{f}^l$ applied by the left (blue) end-effector is shown with the middle plot. The knots of the trajectory with resolution 3 are shown in the bottom graph along with the contact distance d of the left end-effector to the object's surface. The contact knots are grey, the swing knots are pink and the pre-contact knot is cyan. It is worth mentioning that all the quantities shown here are optimized, except the order of phases.

Phase-specific constraints: We extend the phase-based parameterization of the motion [24], considering three possible states between two rigid bodies as described in [28], and we split the knots in three sets called phases: the contact, swing, and pre-contact sets, shown in Fig. 3. In each discrete point of the motion (knot) a constant subset of constraints needs to be satisfied, allowing most of the phase-specific constraints to be time independent. This allows to optimize each phase's duration and satisfy the constraints of each phase simultaneously. Also, each phase is characterized by a distinct set of decision variables that allows us to enforce a number of constraints implicitly, resulting to a reduced number of the decision variables. A list of the constraints categorized according to the phase of the motion follows.

1. Contact phase:
   - Unilateral forces: ${\mathbf{f}_k^i}^T \mathbf{n}_k^c \leq 0$ (12)
   - Linearized friction cone: $|{\mathbf{f}_k^i}^T \mathbf{t}_k^c| \leq \mu {\mathbf{f}_k^i}^T \mathbf{n}_k^c$ (13)
   - Contact point is not slipping (*Implicit constraint*): $\dot{\mathbf{c}}_k^i = 0.$ (14)
2. Swing phase:
   - Continuity of end-effector's motion: $\mathbf{c}_{k+1}^i = f_m(\mathbf{c}_k^i, \dot{\mathbf{c}}_k^i, \Delta \mathbf{T}_k)$ (15)
   - End-effector's swing motion away from object: $d(\mathbf{c}_k^i, \mathcal{S}_{obj}(\mathbf{y}_k, \mathbf{c}_k^i)) > 0$ (16)
   - No force (*Implicit constraint*): $\mathbf{f}_k^i = 0.$ (17)
3. Pre-contact phase:
   - End-effectors touching the object: $d(\mathbf{c}_k^i, \mathcal{S}_{obj}(\mathbf{y}_k, \mathbf{c}_k^i)) \approx 0$ (18)
   - No force (*Implicit constraint*): $\mathbf{f}_k^i = 0,$ (19)

where $\mathbf{n}_k^c \in \mathbb{R}^\nu$ is the normal and $\mathbf{t}_k^c \in \mathbb{R}^\nu$ is the tangent vector at the contact point on the object's surface, $\mu \in \mathbb{R}$ is the friction coefficient, d is a function that computes the Cartesian distance between two points in space, and $\mathcal{S}_{obj}$ computes the location on the object's surface which has the minimum distance to a given point in space.

Input variables and hyper-parameters: The only required input variable is the description of the manipulation task, M_{task}. This is the start and goal state of the object, denoted as $[\mathbf{y}_0^*, \dot{\mathbf{y}}_0^*]$ and $[\mathbf{y}_K^*, \dot{\mathbf{y}}_K^*]$, respectively. To solve the TO problem three hyper-parameters need to be specified. i) The resolution of the mesh $r \in \mathbf{Z}^+$ shown in Fig. 3. ii) The maximum total duration of the generated trajectory T_K. iii) The arm transition matrix $\mathbf{H} \in \{0,1\}^{\iota \times K}$, that only specifies the arm synchronization pattern, i.e. the order with which the arms change contact locations.

Object representation: The object's surface is represented with a closed spline curve as an alternative to [29]. The spline representation is a smooth description of the object's surface from which all relevant properties along with their derivatives can be extracted, like normal and tangent vectors.

3.2 Dyadic planning through trajectory optimization:

Up to this point, we have formulated an optimization problem to generate motion plans in hybrid spaces for an agent acting alone. However, in DcM scenarios by definition the object is jointly manipulated by both individuals. Having the core structure of the optimization defined above, it is straightforward to incorporate the partner's policy in the proposed method. The *collocation* constraints defined in Eq. (7) are now subject also to the partner's forces and can be determined according to the following equations:

$$m\ddot{\mathbf{y}}_k^l + m\mathbf{g} = \sum_{i=0}^{\iota} f_k^i + \boldsymbol{\lambda}_f \qquad (20) \qquad\qquad \mathbf{I}\ddot{\theta}_k + \dot{\theta}_k \times \mathbf{I}\dot{\theta}_k = \mathbf{f}_k^i \times \mathbf{c}_k^i + \boldsymbol{\lambda}_\tau, \qquad (21)$$

where $\mathbf{g}$ is the gravitational acceleration, $\ddot{\mathbf{y}}_k^l$ denotes the translational acceleration of the object's pose vector at the k$_{th}$ knot, $\mathbf{I} \in \mathbb{R}^{\nu\times\nu}$ and $m \in \mathbb{R}$ are the inertia matrix and the mass of the object, respectively. $\boldsymbol{\lambda}_f, \boldsymbol{\lambda}_\tau$, are the force and the torque of the partner's applied wrenches $\boldsymbol{\lambda}$, which are generated from the partner's policy. By updating the *collocation* constraint Eq. (7) according to the new dynamics, the optimization process generates plans that take into account the policy of the partner. As such the object's motion prediction mentioned in Fig. 2 is realized. In contrast to [30], where the method assumes full control authority over the partner's actions, here the only requirement is a model of the partner's policy, which is described below.

Partner's policy model: This work is not focused on the modelling of the partner's policy π_p, but aims to provide a principled way towards incorporating partner's actions into the policy of the agent. We use here a very simple model for the partner's policy, described with Eq. (22). The parameters $\mathbf{K}^p, \mathbf{D}^p \in \mathbb{R}^{\nu\times\nu}$ denote a spring-damper behaviour of the partner towards the goal $[\mathbf{y}_K^*, \dot{\mathbf{y}}_K^*]$ of the co-manipulation task. $\mathbf{K}^p$ can be interpreted as the parameter that can shape whether the partner acts as a leader $\mathbf{K}^p \gg 0$ or as a follower $\mathbf{K}^p = 0$, along with all the intermediate behaviours in between. This model allows us to capture the essence of the dyadic planning problem and to show in the next section the practical importance of the proposed formalism and method.

$$\boldsymbol{\lambda} = \mathbf{K}^p(\mathbf{y}_K^* - \mathbf{y}_k) + \mathbf{D}^p(\dot{\mathbf{y}}_K^* - \dot{\mathbf{y}}_k) \qquad (22)$$

4 Experimental Results

We experimentally validate the proposed method with both a single agent and a dyadic setup [1]. The state of the object is $y_k = [x_k, z_k, \theta_k]$ and the task dimension $\nu = 2$. For motion plans with a single change of contact per arm the total number of decision variables is 177 to 579 depending on the resolution used. These problems are solved within 3 to 80 seconds with an unoptimized MATLAB (fmincon, interior-point) implementation. No special care was taken regarding the initialization of the optimizer.

Res	X(m)	Y(m)	θ(deg)	Time(s)
2	0.034	0.125	8.42	0.25
4	0.027	0.013	0.92	0.43
8	0.023	0.013	1.32	0.72

Table 1: RMSE values against three different mesh resolutions, for 0.5 meter translation and 90° rotation task including a single contact change per arm.

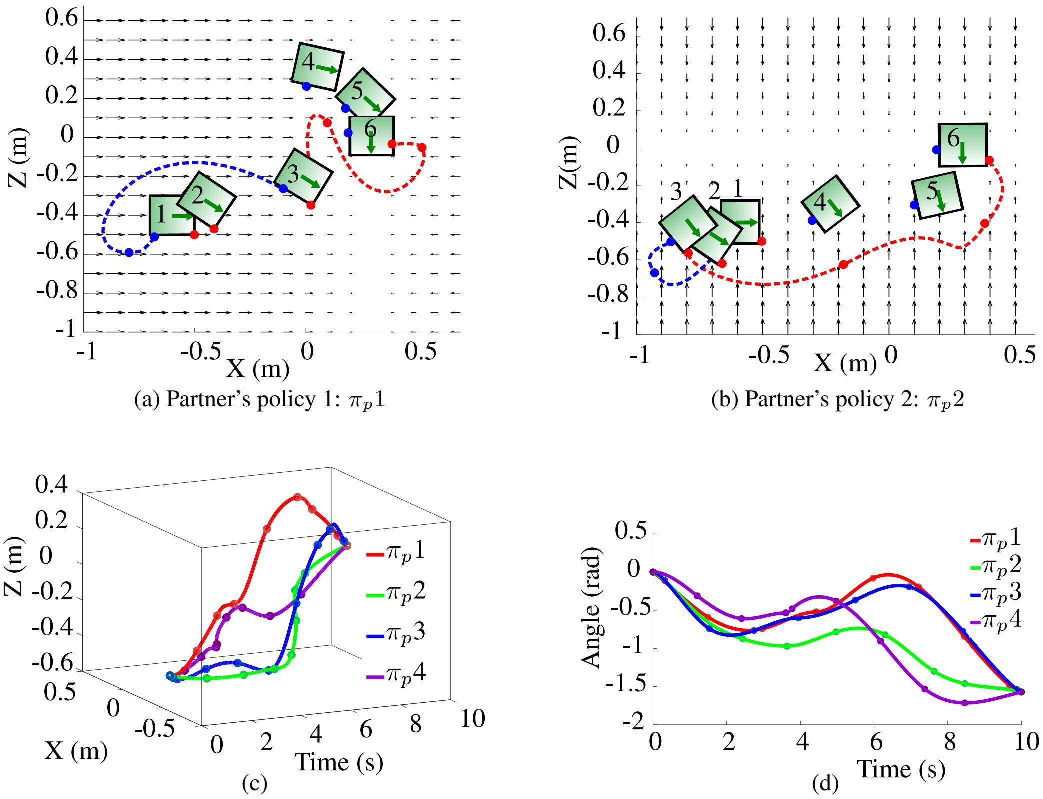
Single agent setup: First, to validate the design choices presented in 3.1, we evaluate the dynamic consistency and the scalability of the generated single agent hybrid plans, in a task of 0.5 meter translation and 90° rotation. Second, to demonstrate the extent of the proposed method's capabilities, we illustrate with Fig. 6(a-f) in simulation a challenging task of throwing and catching a ball.

Evaluation of dynamic accuracy:
We compare the object's trajectory generated by the proposed method and the trajectory generated by a Simulink-based dynamic simulation after feed-forward streaming the planned forces to it. Treating the dynamic simulation as ground truth we compute the root mean square error (RMSE). Table 1 shows the RMSE between the two trajectories along each dimension of the motion, as well as the required computation time per iteration of the optimizer. For a resolution of 2, the simulated trajectory diverges from the planned, revealing the need for higher resolution. However, by comparing resolutions 4 and 8, we can observe that it is not always the case that the higher the resolution the better the accuracy. Thus, sensible resolution must be chosen for each setup.

# Contacts	Vars	Time(s)
1	311	0.43
2	549	0.55
3	787	0.88
4	1025	1.77

Table 2: The number of decision variables and computation times per iteration with respect to the number of contact changes per arm. The task is 0.5 meter translation and 90° rotation.

[1]For video footage of the simulations and the human-robot experiments during DcM tasks, see: https://public.3.basecamp.com/p/uc8DDU9EDDrgmoU11TGLRkKk



(a) Partner's policy 1: π_p1

(b) Partner's policy 2: π_p2

(c)

(d)

Figure 4: In (a) and (b) generated motion plans in response to two different partner policies are illustrated. The green rectangle is the manipulated object, the red and blue dots are the right and left end-effectors of the agent. The start pose is annotated with 1 and the goal with 6. The arrow field illustrates the forces applied by the partner. The key point is that most of the object's trajectory is planned at the active regions of the partner's force field, indicating that the robot utilizes the partner's contribution to the task accordingly. Individual dimensions of the motion trajectories are displayed separately against time for four distinct partner's policies, i.e., x and z dimensions in (c) and θ dimension in (d).

Evaluation of curse of dimensionality: Towards evaluating the scalability of the proposed method in terms of contact changes, the same task is solved with 1, 2, 3 and 4 contact changes per arm. The resolution used is 4 as it was the most prominent for the task and Table 2 provides the respective quantities of interest. Computation times scale linearly as the number of contact changes increases, however large number of unnecessary contact changes hinders the performance of the method. This aspect of the method is particularly important in comparison to mixed-integer approaches [31, 32], that need to explore both the continuous and combinatorial part of the problem, which is prohibitively expensive as the number of optimization variables increases.

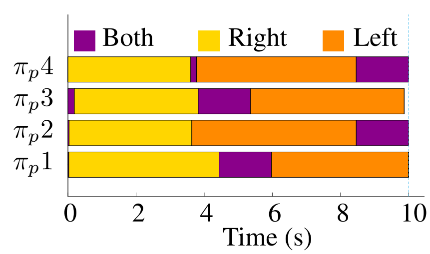


Figure 5: The arm synchronization pattern for the four distinct partner's policies shown in Figs. 4c and 4d. The colours indicate which arm is in contact with the object.

Dyadic setup: First, simulations demonstrate the capabilities of the proposed method to create partner-tailored motion plans in DcM scenarios with respect to different partner's policies. Second, we validate our approach in a real setting with a human partner jointly manipulating an object, along with a bi-manual $\iota = 2$ and $n = 32$ DoF robot.

Experiments with different partner behaviours: We demonstrate the capabilities of the proposed method to adapt the resulting solution with respect to variability of the partner's policy. All inputs

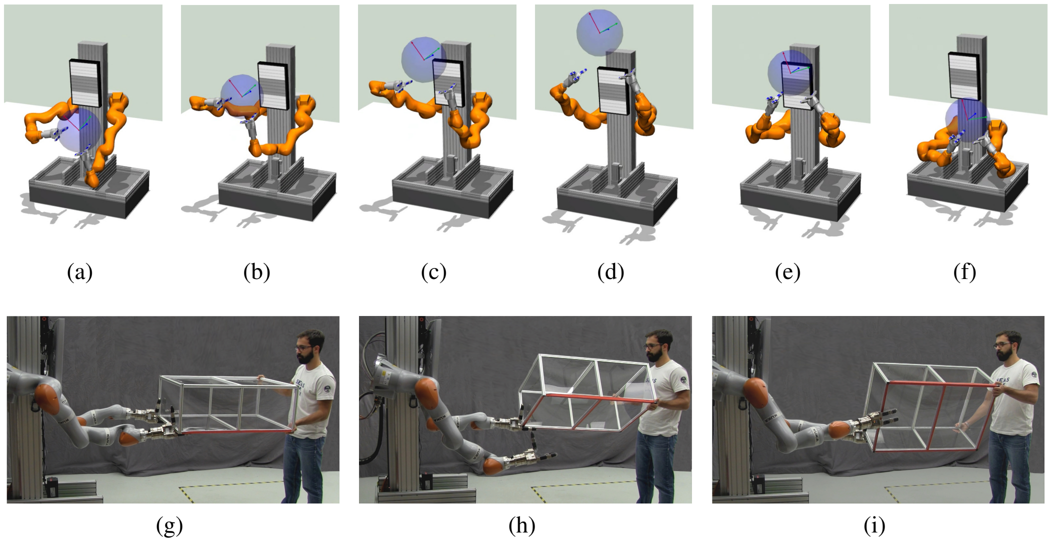


Figure 6: Upper row are key-frames of a single agent hybrid motion and lower row are key-frames of a DcM scenario between the human and the robot.

and hyper-parameters are fixed, while the partner's policy properties, $\mathbf{K}^p$, $\mathbf{D}^p$ in Eq. (22), are altered. Fig. 4a and Fig. 4b illustrate solutions for 0.98 meter translation and $-90°$ rotation task, generated as responses to two different partner policies. In order to make more evident to the reader the variation of the computed solutions, in Figs. 4c, 4d and 5 we present quantitative plots of important trajectory quantities, for four distinct partner's policies. Each partner's policy is illustrated as a force field along one axis: $\pi_p 1$ along X axis, $\pi_p 2$ along Z axis, $\pi_p 3$ along the diagonal between XZ axes and $\pi_p 4$ along θ axis. Although the partner's model described in Section 3.2 is a simple one, it allows us to evoke a variety of different partner policies and demonstrate partner-tailored hybrid robot policies.

Experiments with robot agent and human: We demonstrate a $90°$ rotation task with one contact change per arm. The hybrid motion plans are optimized in the task space as described in Section 3 and are realized on the robot in a feed-forward fashion, after being mapped to the configuration space using inverse kinematics. A more detailed description of the physical system can be found in [33]. Further, the robot policy is realized in the 6D space by utilizing multi-finger grasp-holds around the planned contact locations as a form of mechanical feedback. The resulting key-frames of a DcM scenario are depicted in Fig. 6. The experiments shown in the video with the human partner acting as leader and as follower exhibit the capabilities of the proposed method to create functional hybrid motion plans, applicable to real human-robot systems.

5 Conclusion

This paper presents a novel concept towards robot motion generation for physical human-robot interaction tasks. We propose a formalization towards joint action, based on the assumption that an estimate of the partner's policy exists. Our approach computes the optimal hybrid policy for the robot to complete manipulation tasks as a member of a dyad or alone. The concept only assumes a roughly known model of the partner's policy, a model of the object and the number of contact changes. With this information, our method computes a dynamically consistent and optimal hybrid solution for the i) trajectory of the object, ii) agent's forces, iii) agent's contact locations and iv) respective timings of these actions. The proposed concepts have been evaluated both in simulations and with an actual human-robot dyad. The results and the robot experiments demonstrate that the proposed method has a large potential to be employed in co-manipulation scenarios. Also, during the robot experiments we identified the potential usefulness of non-stationary human policy models, especially for long horizon motions. To this front we are working on an MPC formulation, to generate iteratively real-time hybrid motion plans. Further future work will focus on more elaborate models for the human, and a tighter integration of feedback to realize a robust behavior in interaction.

Acknowledgments

This work is supported by the Honda Research Institute Europe, the Engineering and Physical Sciences Research Council (EPSRC), as part of the CDT in RAS at Heriot-Watt University and The University of Edinburgh, as part of the UK RAI Hub in Future AI and Robotics for Space (FAIR-SPACE) with project ID: EP/R026092/1, and UKIERI as part of the project with ID: UKIERI-DST 2016-17-0152. The authors would like to thank Tamas Bates and Dirk Ruiken for their help with the experiments and Maria Petalidou for the support. Finally, we would like to thank the anonymous reviewers for their suggestions on improving the quality of the paper.

References

[1] S. S. Obhi and N. Sebanz. Moving together: toward understanding the mechanisms of joint action. *Experimental Brain Research*, 211(3):329, May 2011.

[2] T. B. Sheridan. Eight ultimate challenges of human-robot communication. In *6th International Workshop on Robot and Human Communication, in International Symposium on Robot and Human Interactive Communication RO-MAN'97*, pages 9–14. IEEE, 1997.

[3] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib. Progress and prospects of the human–robot collaboration. *Autonomous Robots*, pages 1–19, 2017.

[4] D. J. Agravante, A. Cherubini, A. Bussy, P. Gergondet, and A. Kheddar. Collaborative human-humanoid carrying using vision and haptic sensing. In *International Conference on Robotics and Automation (ICRA)*, pages 607–612. IEEE, 2014.

[5] M. Lawitzky, A. Mörtl, and S. Hirche. Load sharing in human-robot cooperative manipulation. In *International Symposium on Robot and Human Interactive Communication RO-MAN'10*, pages 185–191. IEEE, 2010.

[6] A. Thobbi, Y. Gu, and W. Sheng. Using human motion estimation for human-robot cooperative manipulation. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 2873–2878. IEEE/RSJ, 2011.

[7] C.-M. Huang, M. Cakmak, and B. Mutlu. Adaptive coordination strategies for human-robot handovers. In *Proceedings of Robotics: Science and Systems*, Rome, Italy, July 2015.

[8] G. Maeda, M. Ewerton, G. Neumann, R. Lioutikov, and J. Peters. Phase estimation for fast action recognition and trajectory generation in human–robot collaboration. *The International Journal of Robotics Research*, 36(13-14):1579–1594, 2017.

[9] L. Rozo Castañeda, S. Calinon, D. Caldwell, P. Jimenez Schlegl, and C. Torras. Learning collaborative impedance-based robot behaviors. In *the Twenty-Seventh Conference on Artificial Intelligence (AAAI)*, pages 1422–1428. AAAI, 2013.

[10] D. Vogt, S. Stepputtis, B. Jung, and H. B. Amor. One-shot learning of human–robot handovers with triadic interaction meshes. *Autonomous Robots*, 42(5):1053–1065, 2018.

[11] Y. Maeda, T. Hara, and T. Arai. Human-robot cooperative manipulation with motion estimation. In *International Conference on Intelligent Robots and Systems, (IROS)*, pages 2240–2245. IEEE/RSJ, 2001.

[12] E. Gribovskaya, A. Kheddar, and A. Billard. Motion learning and adaptive impedance for robot control during physical interaction with humans. In *International Conference on Robotics and Automation (ICRA)*, pages 4326–4332. IEEE, 2011.

[13] B. Busch, G. Maeda, Y. Mollard, M. Demangeat, and M. Lopes. Postural optimization for an ergonomic human-robot interaction. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 2778–2785. IEEE, 2017.

[14] L. Peternel, N. Tsagarakis, and A. Ajoudani. A human–robot co-manipulation approach based on human sensorimotor information. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(7):811–822, 2017.

[15] J. Mainprice and D. Berenson. Human-robot collaborative manipulation planning using early prediction of human motion. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 299–306. IEEE, 2013.

[16] H.-C. Lin, J. Smith, K. K. Babarahmati, N. Dehio, and M. Mistry. A projected inverse dynamics approach for multi-arm cartesian impedance control. In *IEEE International Conference on Robotics and Automation*, 2018.

[17] K. Otani, K. Bouyarmane, and S. Ivaldi. Generating assistive humanoid motions for co-manipulation tasks with a multi-robot quadratic program controller. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3107–3113. IEEE, 2018.

[18] E. Noohi, M. Žefran, and J. L. Patton. A model for human–human collaborative object manipulation and its application to human–robot interaction. *IEEE Transactions on Robotics*, 32(4): 880–896, 2016.

[19] M. A. Roa and R. Suárez. Grasp quality measures: review and performance. *Autonomous robots*, 38(1):65–88, 2015.

[20] S. Tonneau, A. D. Prete, J. Pettr, C. Park, D. Manocha, and N. Mansard. An efficient acyclic contact planner for multiped robots. *IEEE Transactions on Robotics*, 34(3):586–601, June 2018.

[21] J. Zhou, R. Paolini, A. M. Johnson, J. A. Bagnell, and M. T. Mason. A probabilistic planning framework for planar grasping under uncertainty. *IEEE Robotics and Automation Letters*, 2 (4):2111–2118, 2017.

[22] J. E. King, J. A. Haustein, S. S. Srinivasa, and T. Asfour. Nonprehensile whole arm rearrangement planning on physics manifolds. In *International Conference on Robotics and Automation (ICRA)*, pages 2508–2515. IEEE, 2015.

[23] M. T. Mason. Toward robotic manipulation. *Annual Review of Control, Robotics, and Autonomous Systems*, (0), 2018.

[24] A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli. Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. *IEEE Robotics and Automation Letters*, 3(3):1560–1567, 2018.

[25] I. Mordatch, E. Todorov, and Z. Popović. Discovery of complex behaviors through contact-invariant optimization. *ACM Transactions on Graphics (TOG)*, 31(4):43, 2012.

[26] M. Kelly. An introduction to trajectory optimization: How to do your own direct collocation. *SIAM Review*, 59(4):849–904, 2017.

[27] M. Posa, C. Cantu, and R. Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research*, 33(1):69–81, 2014.

[28] R. Featherstone. *Rigid body dynamics algorithms*. Springer, 2014.

[29] S. Dragiev, M. Toussaint, and M. Gienger. Gaussian process implicit surfaces for shape estimation and grasping. In *International Conference on Robotics and Automation (ICRA)*, pages 2845–2850. IEEE, 2011.

[30] M. Toussaint and M. Lopes. Multi-bound tree search for logic-geometric programming in cooperative manipulation domains. In *International Conference on Robotics and Automation (ICRA)*, pages 4044–4051. IEEE, 2017.

[31] A. K. Valenzuela. *Mixed-integer convex optimization for planning aggressive motions of legged robots over rough terrain*. PhD thesis, Massachusetts Institute of Technology, 2016.

[32] F. R. Hogan and A. Rodriguez. Feedback control of the pusher-slider system: A story of hybrid and underactuated contact dynamics. *In Proceedings of the 12th International Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2016.

[33] M. Gienger, D. Ruiken, T. Bates, M. Regaieg, M. Meißner, J. Kober, P. Seiwald, and A.-C. Hildebrandt. Human-robot cooperative object manipulation with contact changes. In *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018.