Formalization of Temporal and Spatial Constraints of Bimanual Manipulation Categories

Franziska Krebs and Tamim Asfour

***Abstract*— Executing bimanual manipulation tasks on humanoid robots introduces additional challenges due to inherent spatial and temporal coordination between both hands. In our previous work, we proposed the Bimanual Manipulation Taxonomy, which defines categories of bimanual manipulation strategies based on the coordination and physical interaction between both hands, the role of each hand in the task, and the symmetry of arm movements during task execution. In this work, we build upon this taxonomy and provide a formalization of temporal and spatial constraints associated with each category of the taxonomy. This formalization uses Petri nets to represent temporal constraints and differentiates between relative and global targets. We incorporate these constraints in a category-specific controller to enable reactive adaptation of the behavior according to the respective coordination constraints. We evaluated our approach in simulation and in real-world experiments on the humanoid robot ARMAR-6. The results demonstrate that category-specific constraints can be enforced when needed while maintaining flexibility to accommodate additional constraints.**

I. Introduction

Manipulation remains a challenge in robotics. Compared to unimanual manipulation, bimanual manipulation adds additional challenges due to aspects such as arm coordination strategies, redundancy resolution in closed kinematic chains, self-collision avoidance, and sophisticated force-based control [1]. In our previous work, we introduced a taxonomy for bimanual manipulation that describes different categories of bimanual coordination patterns based on temporal and spatial constraints between both hands [2]. Examples of such bimanual patterns are shown in Fig. 1. The taxonomy distinguishes in total between *seven* patterns of bimanual manipulation actions, see Fig. 2. In this work, we extend and detail the description of our taxonomy and provide a formalization of the temporal and spatial constraints of each category of the taxonomy with a specific focus on linking the taxonomy to controllers needed for the execution of bimanual tasks on a robot. In contrast to previous approaches, this work does not focus on representing a specific bimanual action or a single category of such actions. Instead, it incorporates various configurable constraint templates derived from human demonstrations and can be used to map a wide range of bimanual actions.

The goal of this work is to demonstrate the use of the taxonomy, specifically the information about a bimanual category, to generate appropriate motions for both hands. Thus, we assume that a bimanual manipulation category together with the trajectories of both hands is given. Such information can be extracted from human demonstrations as presented in [2], where we show that the categories can be recognized based on motion capture or RGB-D data [3].

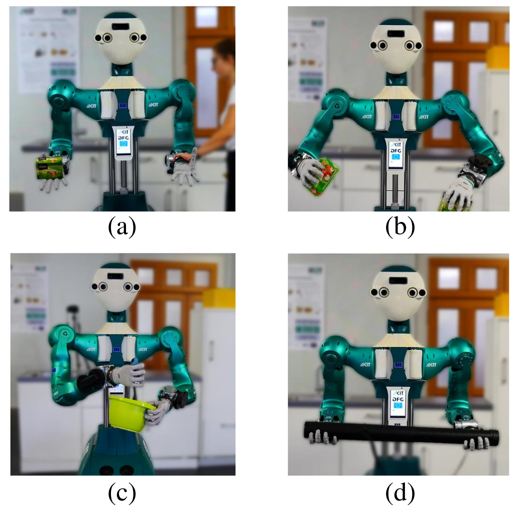


Fig. 1. ARMAR-6 executing actions of different categories: (a) unimanual right, (b) uncoordinated, (c) asymmetric (d) right dominant and symmetric.

Our contribution: Based on the Bimanual Manipulation Taxonomy presented in [2] we formalize the temporal and spatial constraints imposed by each category. This is needed to demonstrate the applicability of the taxonomy to bimanual robot manipulation tasks. Furthermore, we show how the category-specific coordination constraints contribute to the exemplary design of reactive behaviors in bimanual tasks. We evaluated our approach in simulation and real-world experiments using the humanoid robot ARMAR-6 [4].

II. Related Work

First, we give a brief description of the Bimanual Manipulation Taxonomy, see [2] and Fig. 2, since this is pivotal for the remainder of the paper. Further, we discuss how previous works tackled the spatial, temporal and force constraints arising from the coordination between two hands.

A. Bimanual Manipulation Taxonomy

Key aspects for the differentiation between categories defined by the taxonomy are coordination and interaction between the hands, the roles of the hands and symmetry. Firstly, the categories are divided into coordinated and uncoordinated categories. The uncoordinated categories consist

This work has been supported by the Carl Zeiss Foundation under the JuBot project and the German Federal Ministry of Education and Research (BMBF) under the project CATCH-HEMI (01KU2012).

The authors are with the High Performance Humanoid Technologies Lab, Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology (KIT), Germany {franziska.krebs,asfour}@kit.edu

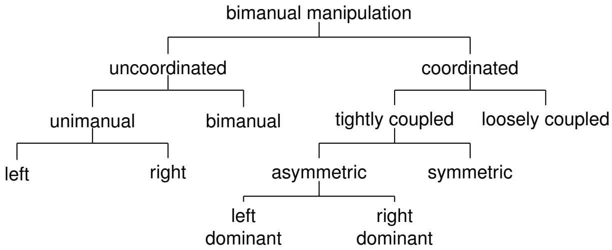


Fig. 2. Bimanual manipulation taxonomy presented in [2].

of uncoordinated bimanual and unimanual left and right. The coordinated categories are further subdivided based on whether there is physical interaction between the hands or not (tightly vs. loosely coupled). Within the tightly coupled categories, the further differentiation depends on whether both hands have the same role when grasping a common object (symmetric) or whether one hand has a dominant behavior (right- or left-dominant). In the following, the abbreviations listed in table I are used for the categories.

TABLE I

ABBREVIATIONS FOR BIMANUAL CATEGORIES.

bimanual category	abbreviation
unimanual left	*uni_l*
unimanual right	*uni_r*
uncoordinated bimanual	*uncoord_bi*
loosely coupled	*loosely*
tightly coupled asymmetrical left dominant	*asym_l_dom*
tightly coupled asymmetrical right dominant	*asym_r_dom*
tightly coupled symmetrical	*sym*

The taxonomy is designed to distinguish between different constraints that apply between hands. At the highest level, the taxonomy distinguishes between coordination and lack thereof among the hands. The uncoordinated categories signify the absence of explicit constraints between the hands. However, it is noteworthy that both arms still share the same workspace, resulting in implicit constraints related to collision avoidance. This could mean prioritizing one over the other, optimizing execution timings to avoid collision, or approaches for motion planning. Conversely, the coordinated categories encompass all coordination patterns wherein temporal and/or spatial constraints might be present. The tightly coupled categories summarize those patterns with physical interaction, which means that additional force constraints are to be considered, while the loosely coupled category represents all temporally and spatially constrained cases without physical interaction.

B. Constraints in Bimanual Manipulation

Several types of constraints can apply between the hands. Pek et al. [5] analyze constraints between multiple objects, distinguish between spatial and temporal constraints and define several subcategories. Another type of constraint commonly considered in robotics is based on forces.

Several approaches exist for describing **temporal constraints**. Allen's Interval Algebra [6] is an established formalism for describing temporal relations on intervals using 13 different relations such as *before*, *during*, etc. Based on this algebra Dreher et al. [7] present a softened formulation of temporal relations which is usable for real-world data and specifically for bimanual manipulation tasks. They further represent temporal task models using graphs and infer subtasks from this representation. Other possibilities to describe temporal constraints are precedence graphs [8] or Petri nets [9]. They can also be directly implemented in the control strategy such as in Mirrazavi et al. [10] who present an approach for coordinating multiple agents to reach a moving object while taking self-collision avoidance into account based on a centralized inverse kinematic solver formulated as a quadratic program.

Spatial constraints can be described on a symbolic level using spatial relations such as *right of*, *behind*, *close* [11], [12] or on trajectory-level. Trajectory-level coordination is mainly addressed by using a leader-follower approach [13], [14] or the cooperative-task space (CTS) [15], [16]. Both focus on physical interaction tasks. Park and Lee [17], [18] propose the extended cooperative-task space (ECTS), a unified representation of both the CTS and the leader-follower approach. Based on two ECTS coefficients, a relative and absolute motion can be distributed to two end-effectors covering 4 different coordination modes: the uncoordinated *orthogonal* mode, the *serial* mode that corresponds to a leader-follower formulation, the symmetrical *parallel* mode, where the hand motions are described relative to an absolute trajectory in between them and finally a *blended* mode. Combined with quadratic programming, the ECTS representation was also recently used for grasping and tossing an object while in motion [19]. Several works use the notion of *global* vs. *relative* task definitions for bimanual manipulation: Describing tasks as a sequence of relative, global and local targets (where local means with respect to the robot frame) [20], representing a task as absolute and relative skills encoded as dynamic movement primitives (DMPs) [21] or proposing a relative parameterization method for bimanual manipulation based on Gaussian Mixture Models (GMMs) [22]. Others introduce a hierarchy where relative goals are prioritized over global ones [23] and deep learning methods that predict when relative actions are required [24]. Gao et al. [25] present a framework for visual imitation learning of bimanual manipulation actions based on hybrid master-slave relationships between the hands and extracted key points on objects. This results in a flexible combination of action definitions both relative to the other hand and the environment, which can automatically be learned.

Force-based constraints between the hands (directly or via tools and objects) play an additional and fundamental role in tightly-coupled categories. The cooperative transport of one object (cf. *sym* in our taxonomy) was extensively investigated in the past. In this case, a certain "squeezing" force needs to be applied in order to keep the object stable between the hands. The employed methods can be categorized into object-level impedance control ([26], [27]) and a combination of Projected Inverse Dynamic Control (PIDC) and the grasp matrix ([28], [29], [30], [31]). Interaction forces can also be relevant constraints for asymmetric tasks as described in [32].

In summary, a wide variety of methods exist to address temporal, spatial and force constraints in bimanual manipulation. They mostly cover specific application scenarios and cannot be generalized to different types of actions. Within this work, we will define general representations for spatial and temporal constraints between the hands which are based on established concepts. We further formulate a template version of those constraints for the different categories of the Bimanual Manipulation Taxonomy. Based on a task that is represented as a sequence of bimanual categories, a specification of temporal and spatial constraints for the complete tasks can be inferred.

III. Formalization of Bimanual Constraints

We propose a formalization of temporal and spatial constraints associated with different categories of our Bimanual Manipulation Taxonomy.

A. Spatial Constraints

Our goal is to provide an approach for the formalization of bimanual constraints that is consistent with the methods known in the literature and discussed in Section II-B. We formalize the spatial constraints imposed on each hand by adopting one of the following states:

- *unspecified:* The hand does not execute a relevant task. Apart from avoiding collisions and fulfilling additional soft criteria such as optimizing the human likeness, the pose of the hand is irrelevant.
- *global:* The goal of the hand is formulated independently of the other hand either in a fixed world-frame or in the robot's root frame.
- *relative:* The goal of the hand is formulated based on the pose of the other hand.

This aligns with the global, local and relative constraints defined by Stavridis et al. [20]. However, in their framework, global constraints refer to a world frame and local constraints to the robot's root frame. In our case, the emphasis is on the constraints between the hands which is why we combine both, and incorporate object-centric task description in the *global* constraints.

TABLE II

Spatial constraints for different categories.

bimanual category	right hand global	right hand relative	left hand global	left hand relative
uncoord_bim	×		×	
uni_l			×	
uni_r	×			
asym_l_dom	×			×
asym_r_dom		×	×	
sym	(×)	×	(×)	×

Table II shows which constraints are most important for each category. The most important task space goals are indicated with an ×, and secondary task space goals with an (×). For the asymmetric categories, this essentially results in a leader-follower approach, where the non-dominant hand is the leader and the dominant hand the follower. The resulting constraints on the end-effector pose target of the hands are described in Section IV. For *uni_r/uni_l* the motion of the right/left hand is determined based on *global* constraints while the respective other hand is *unspecified*. The three states of spatial constraints *unspecified*, *global*, *relative* can apply either for only the start and end point of a segment or the complete trajectory. We formalize the spatial specification as a directed graph where nodes correspond to spatial states and edges to the trajectories connecting them. The types of edges E and nodes N are defined in Fig. 3.

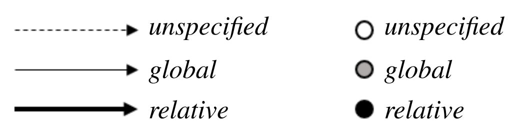


Fig. 3. Edges E (left) and nodes N (right) of the spatial graph.

We consider all possible sets of trajectories $e \in E$ and goal states $n \in N$ in Figure 4.

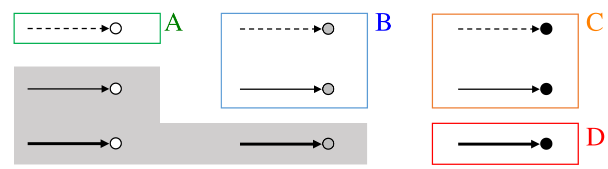


Fig. 4. Spatial permutations considering elements defined in Figure 3. Based on the indicated categories A–D they can be correlated with different bimanual categories.

The permutations highlighted in grey are logically impossible since they are contradictory: If the entire trajectory is defined globally, the goal state cannot be unspecified, and if the trajectory is defined relatively, the goal state must also defined relatively. The remaining permutations are primarily grouped based on the goal pose, denoted by A–D. The combination of a relative edge and node (group D) is treated separately since this case is particularly relevant for actions with physical interaction between the hands (tightly coupled categories). Different combinations of those groups for the left and right hand correspond to different categories of the Bimanual Manipulation Taxonomy. The assignment of spatial combinations to bimanual categories is given Table III), with × indicating that this combination does not exist.

TABLE III

Assignment of spatial combinations to bimanual categories.

Right hand \ Left hand	A	B	C	D
A	*no action*	*uni_l*	*loosely*	*asym_l_dom* or *loosely*
B	*uni_r*	*uncoord_bi*	*loosely*	*asym_l_dom* or *loosely*
C	*loosely*	*loosely*	×	×
D	*asym_r_dom* or *loosely*	*asym_r_dom* or *loosely*	×	× or *sym*

If a task description as a sequence of categories is given, one can construct the representation based on the elements shown in Figure 3. The differentiation between the *unspecified* and *global* trajectories within the blue category (B) and orange category (C), however, require further investigation. In our application in Section IV we use the more restrictive formulation with the *global* trajectory definition. The combinations at the bottom right of the table (indicated with $\times$) do not exist since they would be underspecified with both goal states being relative to the other one. Combinations which are labeled *asym_r_dom* or *asym_l_dom* could also fall into the *loosely* category in case there is no physical interaction (no force constraints) between the hands. The category *sym* is only denoted in brackets in Table III. As shown in Table II for *sym*, relative constraints are most important for both hands. However, since this leaves the system underspecified, some global grounding needs to be introduced. Conceptually, this is possible by defining both hands relative to the same global trajectory. Another option to maintain the relative pose in case of perturbations is to use the asymmetric formulation and make sure that the perturbed arm is considered as leader (non-dominant hand).

B. Temporal Constraints

Zöllner et al. [9] use Petri nets to describe how task executions of the right/left arm or both arms can be triggered. This means switching between an *active* and a *ready* state for both hands. In this work, we formulate the category-specific constraints and behavior in case of perturbation by defining different Petri net templates (see Figure 5). In comparison to alternative approaches such as behavior trees, Petri nets naturally model concurrent and parallel processes. Consequently, they are particularly well-suited for representing shared resources, such as the coordination of two hands in our context. Petri nets are defined by the 4-tuple $N = (P, T, A, mo)$, with P being the set of places, T the set of transitions, A the set of arcs and m_0 the initial marking of the net. In Figure 5 transitions are denoted within the rectangles and places are indicated next to the circles. Petri net templates are defined to describe individual bimanual categories and can be sequenced to formulate the entire task. Initial markings are at the left, indicating their entry from a previous category and their transition to the next one on the right. The category-specific template Petri nets describe, on the one hand, the category-specific behavior in the event of perturbations (phase stopping) and, on the other hand, the behavior during transitions (e. g., transition independently or only when both are completed).

The terms *ErrL* and *ErrR* as part of the transitions T indicate the error of the left and right hand respectively, which is computed based on the current pose and the computed pose in the last timestep. The transition is triggered in case of a certain error surpassing a specified threshold. Since the category *loosely* includes all categories with spatial and temporal constraints but without physical interaction, we do not provide a temporal Petri net template for this category since it combines several different coordination patterns. This might require further investigation and potentially the definition of subcategories within *loosely*.

Spatial and temporal constraints are largely modeled in parallel in this model but there are several dependencies. Each segment (each category) has a formalization of spatial and temporal constraints. The spatial constraints define if a *relative* or *global* trajectory is followed within the Petri nets. The time constraints can stop the progression of these reference trajectories, which in turn has spatial effects.

C. Transitions Between Categories

In the previous subsection, we described how spatial and temporal constraints can be formalized within a bimanual category. However, a task consists of a sequence of those categories and the associated trajectories. In this work, we consider the order of sequences as given. They can be derived from demonstration, e. g., using methods presented in Dreher et al. [7]. The temporal conditions for transitions between categories are implicitly derived from the representation as Petri nets. This includes synchronization at the start of a category such as for *asym_r_dom*/*asym_l_dom* and *sym*. In the case of the uncoordinated categories, the hands evolve independently.

Spatially, several categories can be concatenated by connecting them to form a common graph. An example for the case of rolling dough is shown in Fig. 6. It consists of the following categories: both hands start from a global resting position, the left hand moves an object out of the way (*uni_l*), both hands move to the rolling pin (*uncoord_bi*), both hands hold the rolling pin and roll out the dough (*sym*) and finally the hands release the rolling pin and move back to their resting position (*uncoord_bi*). The symmetric segment is represented as a combination of *relative* and *global* since both are needed for a complete description.

If no specific trajectory is defined, traditional planners can be used. Given trajectories can be easily adapted to a new start or goal poses by using movement primitives such as DMPs [33], ProMPs [34] or VMPs [35].

IV. Category-based Controller

The above-mentioned bimanual categories can be used in robotics by selecting control methods that ensure the category-specific constraints are met. Our aim is to develop an approach suitable for robots acting in the proximity of humans, e. g., assistive robots in a household context. Therefore, a compliant robot behavior is required, which poses the challenge of appropriately reacting to external perturbations.

To this end, we present a framework that guarantees the fulfillment of category-specific spatial and temporal constraints in a reactive manner. The system assumes minimal knowledge of the environment or force-torque sensing. The framework is depicted in Figure 7. The task model includes the sequence of bimanual categories including movement primitives and provides desired relative and absolute hand poses as well as the category label to the category-based motion generation. Based on this information and the current state of the robot, the category-based motion generation

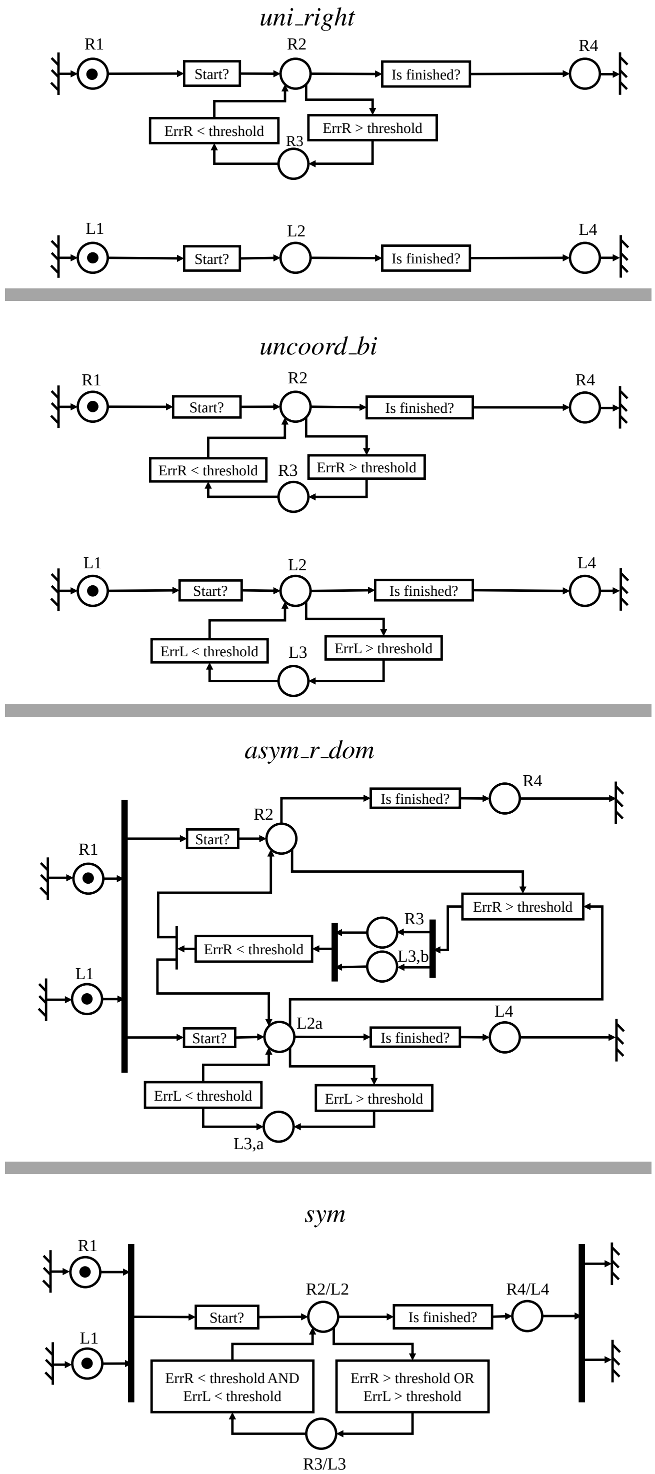


Fig. 5. Temporal constraints with places P for right/left hand respectively: R1/L1: ready, R2/L2: active, R3/L3: paused, R4/L4: completed

targets for the robot robot. These targets are then mapped to robot control commands by using an impedance controller for tracking the desired task space goals.

The category-specific behavior is designed based on the respective constraints imposed on each hand as described in detail in Section III-A. Therefore we consider the categories

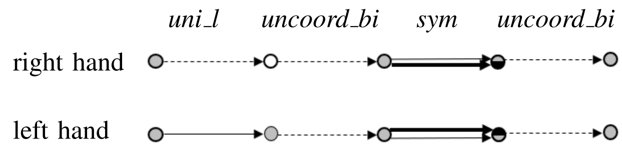


Fig. 6. Temporal sequencing for rolling dough.

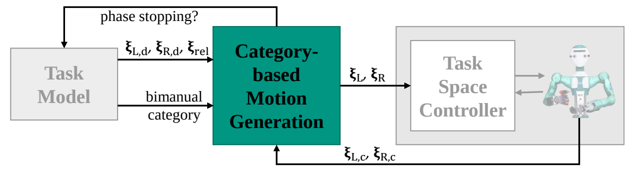


Fig. 7. Framework for category-based robot control. The meaning of variables in indicated in Table IV.

listed in Table II and parameterize their trajectories taking into account the most relevant constraints. As input, the task model provides the currently desired poses of both hands. Leveraging this information, we can formulate reactive target postures for both hands based on the categories. Poses are represented as homogenous 4x4 matrices ξ

which are defined relative to the global coordinate system. Each hand's target pose can be described either directly by their global target pose

$$\xi_L = \xi_{L,d} \qquad\qquad \xi_R = \xi_{R,d} \tag{1}$$

or based on the current pose of the other hand and the relative trajectory. The relative trajectory can be learned based on the initially given, unchanged trajectories. Relative poses ξ_{rel} are defined as the pose of the right hand relative to the left hand. Therefore, global targets based on the relative pose are computed as

$$\xi_R = \xi_{L,c}\xi_{rel} \qquad\qquad \xi_L = \xi_{R,c}\xi_{rel}^{-1}. \tag{2}$$

The variables used in these equations are defined in Table IV.

TABLE IV

GLOBAL VARIABLES FOR THE DESCRIPTIONS OF TASK SPACE GOALS.

variable	meaning
ξ_L, ξ_R	computed target hand pose
$\xi_{L,d}, \xi_{R,d}$	desired hand pose for the current state
ξ_{rel}	desired relative pose (right hand in frame of left)
$\xi_{L,c}, \xi_{R,c}$	actual measured hand pose for the current state

For different categories, the targets are combined in different ways as indicated in Table II. For the case *asym_r_dom* the left hand has the role of the *leader* $\xi_L = \xi_{L,d}$ and the right hand of the *follower* $\xi_R = \xi_{L,c}\xi_{rel}$,

while for the *uncoord_bi* case both are described globally as $\xi_L = \xi_{L,d}$, $\xi_R = \xi_{R,d}$.

The most complex case is *sym*. As shown in Table II, the relative poses are most important, however, if possible also a global pose should be followed. To achieve this, one can alternate between the two asymmetric categories while selecting the perturbed hand as the leader (i. e., the non-dominant hand). This is implemented in a way where switching leader and follower is only triggered if the error computed for the follower surpasses the error computed for the leader by a certain margin.

Temporal constraints are considered by implementing the logic of the Petri net templates presented in Figure 5. This allows uncoordinated motions to progress temporally independently. However, should a perturbation occur, the progression of the respective trajectories, is halted. This aligns with *phase stopping* of movement primitives.

For the tightly coupled categories, a start synchronization is implemented wherein the segment only starts if both hands are ready i. e., if the preceding segment is completed for both hands. In symmetric cases, the motions of the hands are described in a shared system, implying that a perturbation of one hand also triggers a *phase stopping* for the other hand, which implicitly leads to goal synchronization. In asymmetric cases, phase stopping is only applied to both hands if the follower is perturbed. Threshold values need to be adapted according to the underlying task-space controller and its tracking accuracy.

To transform the task space goals into torque commands for the robot, a common impedance controller for task space targets is used. The compliant behavior of this controller ensures safe physical interaction with humans and the environment.

V. Experiments and Evaluation

We evaluate the approach described in Section IV by executing qualitative example tasks on a real robot and performing quantitative evaluations in a Mujoco simulation.

A. Real Robot Experiments

We implement four typical household actions as example scenarios representing different categories and executed them on the ARMAR-6 [4] robot to demonstrate the feasibility of the approach in real world. The four scenarios are shown in Fig. 1. Recordings can be found in the accompanying video.

1) Unimanual: The robot is in a kitchen holding a box of tea bags in its right hand. The left hand is blocking a drawer, but the human is able to guide the unspecified hand out of the way to access the drawer.

2) Uncoordinated: The robot holds a box of tea in each hand and places them on a table. In Figure 8(a), it can be observed that the placing motion performed by the left hand is continued even if the other hand is stopped. The offset between the actual and the desired position during perturbation corresponds to the threshold set for the detection of perturbations.

3) Asymmetric: The robot holds a bowl in its left hand and stirs with a ladle held by the right hand. The stirring motion is continued with respect to the pot even if the hand with the bowl is perturbed (see Fig. 8(b)).

4) Symmetric: The robot grasps a long tube and moves it upwards. First, the left and afterwards the right hand is perturbed during the motion (see Fig. 8(c)). Since the controller is initialized as the right hand following the left, the right hand directly adapts its motion when the left hand is perturbed. As soon as a threshold for the difference between the desired and actual pose is reached, the progression of the trajectory stops until the hand is released. When the right hand is perturbed afterwards, a critical threshold needs to be reached to change the behavior into the left hand following the right one. Once again phase stopping occurs.

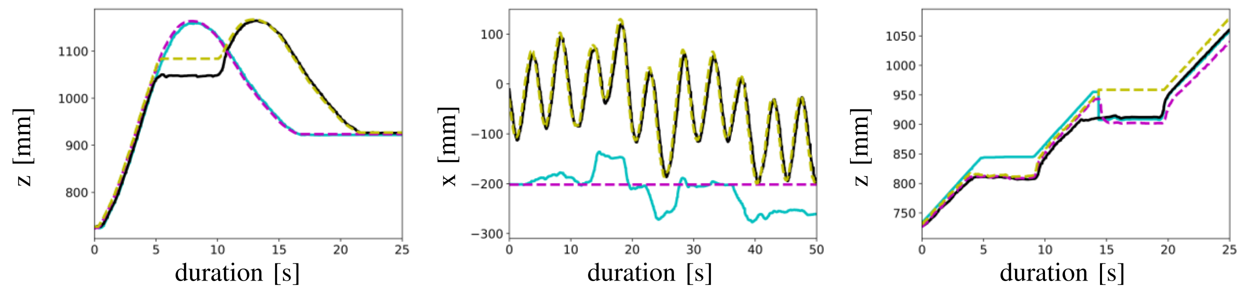
Overall, the robot behaves as expected for the respective tasks: Relaxed coordination constraints enable the consideration of additional constraints such as moving out of the way *unimanual* and avoiding unnecessary delays of the left hand *uncoordinated*. In the *asymmetric* example, the task can be continued even when perturbed and for the *symmetric* case, the tube can be reliably held.

B. Simulation Experiments

To quantitatively evaluate how well the presented approach ensures to fulfill spatial constraints, we perform experiments on the humanoid robot ARMAR-6 [4] in a Mujoco simulation [36]. Experiments in simulation enable us to apply the perturbation in a reproducible way, which allows for comparable results. As shown in Fig. 9 the robot performs a simple motion downwards with a constant offset between the hands. The total duration of the motion is $10sec$. For time $t \in [3; 7]$ seconds, a perturbation force is applied on the left arm (indicated by the red arrow, in negative z direction, 100 Newton). We set the controller parametrization according to the bimanual categories. Fig. 10 shows the positions for three different cases. In the *uncoord_bim* case, the motion of the right hand is not influenced by the perturbation of the left hand. For *asym_l_dom*, the right hand does not adapt spatially but stops its execution. Lastly, for *sym*, the right hand does not only adapt spatially but also stops the execution of its trajectory as long as the left hand is perturbed. Fig. 10 shows also a perturbation in the x- and y-direction since the hand is perturbed at the wrist but the position is measured for the tool center point of the robot, which is located close to the center of the palm.

Each experiment is performed 5 times with different simulation seed values. We measure the absolute and relative errors of the hands for both the unperturbed case and the perturbed case. Errors are computed as the translational error between two frames, i. e. the error between e.g. $\xi_{\mathbf{R},\mathbf{d}}$ and $\xi_{\mathbf{R},\mathbf{c}}$ as the absolute error and between $\xi_{\mathbf{L},\mathbf{d}}^{-1}\xi_{\mathbf{R},\mathbf{d}}$ and $\xi_{\mathbf{rel}}$ as the relative error.

The average errors are shown in Table V. The most relevant errors for each category are highlighted in gray. It can be observed that the highlighted errors have mostly smaller values compared to the other errors in the same row. This is because the approach prevents the accumulation of global errors which could for example negatively impact the relative error. For the perturbated case for all categories, the absolute error for the left hand is high because the hand is perturbed and can therefore not be influenced by our approach. When taking a look at the highlighted errors for the absolute error of the left hand and the relative error they are mostly small which indicates, that the respectively relevant constraints are enforced. Only for the *asym_l_dom*



(a) Perturbation of the right hand at t = 5s for an *uncoordinated* reaching motion.

(b) Perturbation of the left hand during an *asym_right_dom* stirring motion.

(c) Perturbation of the left hand at t = 4s and of the right at t = 13s for *symmetric*.

Fig. 8. Real-world experiments performed on ARMAR-6. Color legend: actual right (—), desired right (- -), actual left (—), desired left(- -).

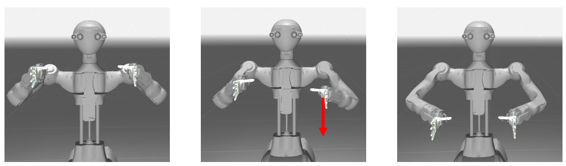


Fig. 9. Robot experiments in a Mujoco simulation. The hands move downwards with a constant offset. For $t \in [3; 7]$s a constant force of 100 N in z-direction is applied on the left wrist.

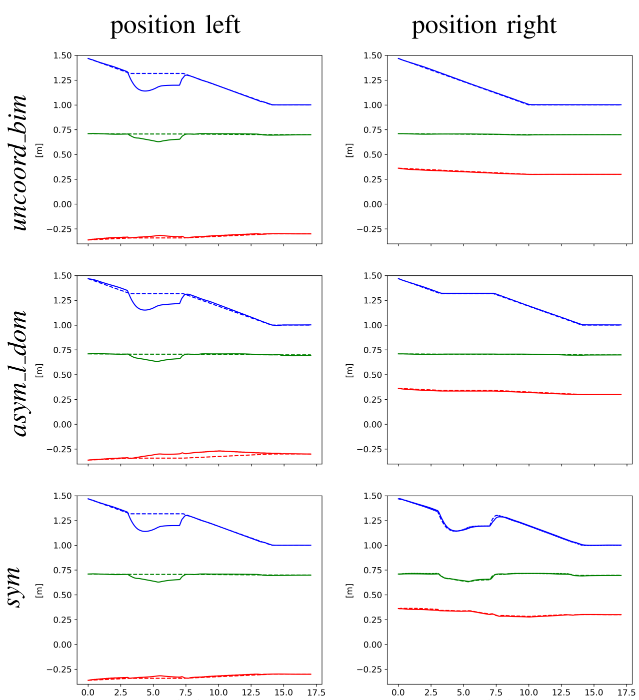


Fig. 10. Positions for both hands in the case of a perturbation of the left hand for $t \in [3; 7]$. Color legend: actual x (—), actual y (—), actual z (—), computed x(- -), computed y(- -), computed z(- -).

case, the relative error is high. This is because $e_{a,R}$ and e_{rel} are conflicting for a perturbation of the left hand, which is why this behavior is expected.

VI. Discussion

In this study, we focus on the control alternatives specific to individual categories, along with their respective features. Although we touch upon how these category-specific definitions facilitate sequencing (see Section III-C), we do not present a comprehensive framework or evaluation. However, the evaluations presented in Section V illustrate the potential advantages of employing category-specific controllers in extended sequences as discussed below.

Two baselines can be used to compare with (i) independent control of the two arms (as in *uncoord_bi*, baseline A) or (ii) a permanent coupling of both arms (as in *sym*, baseline B). The categories can be concatenated as described in Section III-C. First, the approaches differ in their execution time when implementing phase stopping in the event of an error. For baseline B, the hands in each segment take the time of the slower hand (due to target synchronization), resulting in a long, inefficient execution time. In contrast, in baseline A, each hand takes only its minimum required time for each segment, resulting in a low execution time but potentially compromising important timing constraints. Our approach ensures that timing constraints are enforced where necessary, but relaxes them where possible to allow for short execution time. The other important criterion for evaluating the approach is the task success. For tasks that require coordination between hands, baseline A performs poorly because relative constraints between hands are not taken into account. Baseline B, on the other hand, enforces relative constraints, but possibly at the cost of global constraints and at the cost of degrees of freedom that could be used for further constraints, e. g., to enforce human-like movements. Our approach aims to enforce the constraints where they are needed, ensuring that the specific relative constraints are met, but keeping the task as free as possible so that additional criteria can be enforced.

TABLE V

Average errors in mm of 5 repetitions performed in Mujoco simulation.

	unperturbed			perturbed		
	$e_{a,L}$	$e_{a,R}$	e_{rel}	$e_{a,L}$	$e_{a,R}$	e_{rel}
uncoord_bim	1.93	1.93	0.30	37.99	1.46	78.35
asym_l_dom	3.69	1.93	1.77	50.02	1.48	35.45
asym_r_dom	1.93	3.69	1.94	37.94	78.96	3.40
sym	1.93	3.69	1.94	37.94	51.68	3.40

VII. Conclusion

In this work, we define and formalize the temporal and spatial constraints that apply to actions within the different categories defined in the Bimanual Manipulation Taxonomy

[2]. We show how these category-specific constraints can be implemented to select suitable robot controllers and evaluate the implemented controllers on the humanoid robot ARMAR-6 [4]. The framework may require further fine-tuning to the task at hand but can be set up quickly and intuitively for an initial task description.

In the future, we plan to evaluate the sequencing of multiple categories as described in Section III-C for the execution of daily life tasks by a robot. In addition, we plan to go beyond the temporal and spatial constraints and also consider force-related constraints. In the context of our taxonomy, this means to take a closer look into coordinated, tightly-coupled manipulation. This includes challenges related to the recognition of force-based constraints and force-based control methods.

References

[1] C. Smith, Y. Karayiannidis, L. Nalpantidis, X. Gratal, P. Qi, D. V. Dimarogonas, and D. Kragic, “Dual arm manipulation—a survey,” *Robotics and Autonomous Systems*, vol. 60, no. 10, pp. 1340–1353, 2012.

[2] F. Krebs and T. Asfour, “A bimanual manipulation taxonomy,” *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 031–11 038, 2022.

[3] F. Krebs, L. Leven, and T. Asfour, “Recognition of bimanual manipulation categories in RGB-D human demonstration,” in *IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2023.

[4] T. Asfour, M. Wächter, L. Kaul, S. Rader, P. Weiner, S. Ottenhaus, R. Grimm, Y. Zhou, M. Grotz, and F. Paus, “Armar-6: A high-performance humanoid for human-robot collaboration in real world scenarios,” *IEEE Robotics & Automation Magazine*, vol. 26, no. 4, pp. 108–121, 2019.

[5] C. Pek, G. F. Schuppe, F. Esposito, J. Tumova, and D. Kragic, “Spatial: monitoring and planning of robotic tasks using spatio-temporal logic specifications,” *Autonomous Robots*, vol. 47, no. 8, pp. 1439–1462, 2023.

[6] J. F. Allen, “Maintaining knowledge about temporal intervals,” *Communications of the ACM*, vol. 26, no. 11, pp. 832–843, 1983.

[7] C. R. G. Dreher and T. Asfour, “Learning temporal task models from human bimanual demonstrations,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2022, pp. 7664–7671.

[8] M. Pardowitz, S. Knoop, R. Dillmann, and R. D. Zöllner, “Incremental learning of tasks from user demonstrations, past experiences, and vocal comments,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 2, pp. 322–332, 2007.

[9] R. Zöllner, T. Asfour, and R. Dillmann, “Programming by demonstration: dual-arm manipulation tasks for humanoid robots,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2004, pp. 479–484.

[10] S. S. Mirrazavi Salehian, N. Figueroa, and A. Billard, “A unified framework for coordinated multi-arm motion planning,” *The International Journal of Robotics Research*, vol. 37, no. 10, pp. 1205–1232, 2018.

[11] S. Guadarrama, L. Riano, D. Golland, D. Go, Y. Jia, D. Klein, P. Abbeel, T. Darrell *et al.*, “Grounding spatial relations for human-robot interaction,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013, pp. 1640–1647.

[12] R. Kartmann, Y. Zhou, D. Liu, F. Paus, and T. Asfour, “Representing spatial object relations as parametric polar distribution for scene manipulation based on verbal commands,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2020, pp. 8373–8380.

[13] J. Luh and Y. Zheng, “Constrained relations between two coordinated industrial robots for motion control,” *The International journal of robotics research*, vol. 6, no. 3, pp. 60–70, 1987.

[14] Y. Zhou, M. Do, and T. Asfour, “Coordinate change dynamic movement primitives - a leader-follower approach,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2016, pp. 5481–5488.

[15] M. Uchiyama and P. Dauchez, “Symmetric kinematic formulation and non-master/slave coordinated control of two-arm robots,” *Advanced Robotics*, vol. 7, no. 4, pp. 361–383, 1992.

[16] B. V. Adorno, P. Fraisse, and S. Druon, “Dual position control strategies using the cooperative dual task-space framework,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2010, pp. 3955–3960.

[17] H. A. Park and C. G. Lee, “Extended cooperative task space for manipulation tasks of humanoid robots,” in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2015, pp. 6088–6093.

[18] ——, “Dual-arm coordinated-motion task specification and performance evaluation,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2016, pp. 929–936.

[19] M. Bombile and A. Billard, “Dual-arm control for coordinated fast grabbing and tossing of an object: Proposing a new approach,” *IEEE Robotics & Automation Magazine*, vol. 29, no. 3, pp. 127–138, 2022.

[20] S. Stavridis, P. Falco, and Z. Doulgeri, “Pick-and-place in dynamic environments with a mobile dual-arm robot equipped with distributed distance sensors,” in *IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2021, pp. 76–82.

[21] È. Pairet, P. Ardón, M. Mistry, and Y. Petillot, “Learning and composing primitive skills for dual-arm manipulation,” in *Towards Autonomous Robotic Systems (TAROS)*. Springer, 2019, pp. 65–77.

[22] J. Liu, H. Sim, C. Li, K. C. Tan, and F. Chen, “Birp: Learning robot generalized bimanual coordination using relative parameterization method on human demonstration,” in *IEEE Conference on Decision and Control (CDC)*, 2023, pp. 8300–8305.

[23] S. Tarbouriech, B. Navarro, P. Fraisse, A. Crosnier, A. Cherubini, and D. Sallé, “An admittance based hierarchical control framework for dual-arm cobots,” *Mechatronics*, vol. 86, p. 102814, 2022.

[24] H. Kim, Y. Ohmura, and Y. Kuniyoshi, “Goal-conditioned dual-action imitation learning for dexterous dual-arm robot manipulation,” *IEEE Trans. on Robotics*, 2024.

[25] J. Gao, Z. Tao, N. Jaquier, and T. Asfour, “Bi-KVIL: Keypoints-based visual imitation learning of bimanual manipulation tasks,” *arXiv preprint arXiv:2403.03270*, 2024.

[26] T. Wimböck, C. Ott, A. Albu-Schäffer, and G. Hirzinger, “Comparison of object-level grasp controllers for dynamic dexterous manipulation,” *The International Journal of Robotics Research*, vol. 31, no. 1, pp. 3–23, 2012.

[27] M. Shahbazi, J. Lee, D. Caldwell, and N. Tsagarakis, “Inverse dynamics control of bimanual object manipulation using orthogonal decomposition: An analytic approach,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2017, pp. 4791–4796.

[28] I. D. Walker, R. A. Freeman, and S. I. Marcus, “Analysis of motion and internal loading of objects grasped by multiple cooperating manipulators,” *The International journal of robotics research*, vol. 10, no. 4, pp. 396–409, 1991.

[29] H.-C. Lin, J. Smith, K. K. Babarahmati, N. Dehio, and M. Mistry, “A projected inverse dynamics approach for dual-arm cartesian impedance control,” *arXiv preprint arXiv:1707.00484*, 2017.

[30] N. Dehio, J. Smith, D. L. Wigand, G. Xin, H.-C. Lin, J. J. Steil, and M. Mistry, “Modeling and control of multi-arm and multi-leg robots: Compensating for object dynamics during grasping,” in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2018, pp. 294–301.

[31] J. Gao, Y. Zhou, and T. Asfour, “Projected force-admittance control for compliant bimanual tasks,” in *IEEE/RAS International Conference on Humanoid Robots (Humanoids)*, 2018, pp. 607–613.

[32] L. Pais Ureche and A. Billard, “Constraints extraction from asymmetrical bimanual tasks and their use in coordinated behavior,” *Robotics and Autonomous Systems*, vol. 103, pp. 222–235, 2018.

[33] A. Ijspeert, J. Nakanishi, and S. Schaal, “Learning attractor landscapes for learning motor primitives,” *Advances in neural information processing systems*, vol. 15, 2002.

[34] A. Paraschos, C. Daniel, J. Peters, G. Neumann *et al.*, “Probabilistic movement primitives,” *Advances in neural information processing systems*, 2013.

[35] Y. Zhou, J. Gao, and T. Asfour, “Learning via-point movement primitives with inter- and extrapolation capabilities,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2019, pp. 4301–4308.

[36] E. Todorov, T. Erez, and Y. Tassa, “Mujoco: A physics engine for model-based control,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2012, pp. 5026–5033.