

Distributed Virtual Model Control for Scalable Human-Robot Collaboration in Shared Workspace

Yi Zhang, Omar Faris, Chapa Sirithunge, Kai-Fung Chu, Fumiya Iida and Fulvio Forni

Abstract—We present a decentralized, agent agnostic, and safety-aware control framework for human-robot collaboration based on Virtual Model Control (VMC). In our approach, both humans and robots are embedded in the same virtual-component-shaped workspace, where motion is the result of the interaction with virtual springs and dampers rather than explicit trajectory planning. A decentralized, force-based stall detector identifies deadlocks, which are resolved through negotiation. This reduces the probability of robots getting stuck in the block placement task from up to 61.2% to zero in our experiments. The framework scales without structural changes thanks to the distributed implementation: in experiments we demonstrate safe collaboration with up to two robots and two humans, and in simulation up to four robots, maintaining inter-agent separation at around 20 cm. Results show that the method shapes robot behavior intuitively by adjusting control parameters and achieves deadlock-free operation across team sizes in all tested scenarios.

I. INTRODUCTION

Human-robot collaboration (HRC) aims to bridge the gap between human dexterity and robot precision. Compared to multi-robot collaboration, HRC in shared workspaces must account for (i) unpredictability and complexity of human motion and (ii) life-critical cost of failure and injury [1]. Hence, HRC literature mainly focuses on ensuring human safety via rich sensing [2], human intent prediction [3], and optimization and learning-based algorithms [4]–[6]. Nonetheless, these methods can be computationally expensive, model- or data-dependent, and tailored mainly to humans.

In shared workspaces, agents should be able to enter and leave freely, sharing workloads and roles. However, current approaches often treat humans and robots differently, applying unique rules for each, with robots typically adapting to human actions [7], [8]. In this paper, we challenge this distinction. Our key question is: can we move towards a lightweight, agent-agnostic, and scalable human-robot collaboration while guaranteeing human safety?

We start from the idea that robot and human agents can have shared or complementary collaborative roles. We search for a decentralized control architecture that can shape the collective behavior without relying on accurate models or large datasets. The architecture should have a low computational load and seamlessly integrate new agents without the need to

Y. Zhang and O. Faris are supported by the Engineering and Physical Sciences Research Council [EP/S023917/1]. This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 101034337. The authors are with the Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, United Kingdom {yz892, of292, csh66, kfc35, fi224}@cam.ac.uk, f.forni@eng.cam.ac.uk

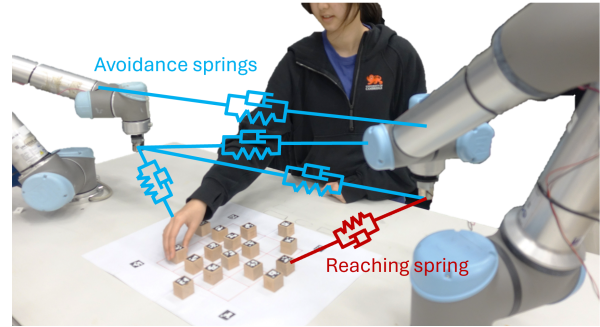


Fig. 1: Human-robot collaboration in a pick and place task.

redesign the control strategy. The goal is to derive a strategy that allows for any number of agents to participate in the task, or withdraw from it, as required, while guaranteeing a safe multi-agent collaboration of human or robots.

For this purpose, we propose a safety-aware control framework based on Virtual Model Control (VMC). In our approach, both humans and robots are embedded in a common workspace shaped with virtual components. Robot motion is regulated by virtual springs and dampers, without explicit trajectory planning. By adjusting virtual component parameters, we can vary the interaction behavior from fast to slow and change how cautious of other agents. Human's behavior is then influenced through their natural adaptation to the workspace. With nonlinear virtual springs, we bound the force applied by the robot upon potential physical interaction.

In this paper, we validate our approach in collaborative pick-and-place task involving both humans and robots, as seen in Fig. 1. Our contributions include

- An **agent-agnostic approach to multi-agent collaboration** where robots and humans are treated on the same ground. The robot behavior and its interaction with other robots and human are shaped by intuitive mechanical parameters. A separation distance of about 20 cm is maintained through active avoidance and compliance.
- A **VMC-based implementation with conflict resolution**, capable of detecting deadlocks via force balance and negotiating priority through minimal communication, eliminates the failure mode of robots getting stuck during the block placement task, reducing its occurrence from a maximum of 61.2% to zero experimentally.
- **Scalability**: the decentralized approach allows seamlessly swapping of agents and smooth adaptation to changes in team composition, shown for up to two robots and two humans in experiment, and up to four robots in simulation.

II. RELATED WORKS

A. Virtual Model Control

VMC offers an intuitive control framework where virtual mechanical components (e.g., springs and dampers) are used to design physics-driven predictable robotic motion. The relevance of VMC for this paper is that it moves beyond pre-programmed or planned motion. Instead, robot behavior is a direct consequence of its interaction with virtual mechanical components. This enables a novel design approach that is particularly well-suited for interactive tasks involving a variety of low-predictability scenarios. VMC was originally introduced for bipedal locomotion in [9] and later adopted for quadruped locomotion [10], [11]. Recently, VMC was utilized for robot manipulation scenarios such as reaching under uncertainties [12], robot assisted laparoscopic surgery tasks [13], and robotic cutting [14].

In HRC, VMC was used in collaborative surgical bone drilling task for increased drill alignment accuracy [15]. In this work, we expand the use VMC towards multi-agent scenarios involving humans and robots. Additionally, we present the use of VMC in position-controlled robotic manipulators, which have not been addressed before. With VMC, all agents can be abstracted as virtual mechanisms, resulting in a decentralized control architecture that enables swift addition or removal of any agent. Agents avoid each other with mechanisms like virtual repulsive springs and dampers. No path planning is required as trajectories emerge from these interactions. This also enables sparse communication among the robots, which is only truly required to handle stalls, detected through force-based analysis.

B. Human-Robot Collaboration

Safety in the shared workspaces is commonly addressed through motion planning guided by sensing and human intention prediction [4], where robots slow down, re-route, or completely stop based on the human state. Researchers in [16] predicted human arm pose to schedule robot goals with a learning-based method in a collaborative pick-and-place task. Compliance with ISO/TS 15066 was addressed in [17] with an online optimizer that generated collision-free trajectories. Potential fields with a reference governor were utilized for real-time safety in [18]. Proximity sensing on the robot body is utilized for contact anticipation and avoidance in [19]. Fast distance evaluation is investigated in [20] for avoiding moving obstacles. Nonetheless, in close proximity during HRC, collision remains possible, and stiff robot behaviors can cause injury in such cases.

Impedance control [21] enables compliant interaction in uncertain environments and has been applied in shared workspaces to mitigate impact forces. It has been combined with proximity sensing for safe, collision-free close interactions [22], with potential fields for obstacle and human avoidance in assembly tasks [23], and with human-in-the-loop adaptation using stress and attention signals to generate personalized trajectories [24].

The VMC of this paper and impedance control are similar in that both methods impose a desired dynamic behavior, like

that of a spring-damper system. The main difference is that VMC unifies motion planning and control by defining the robot's end-effector trajectory through its interaction with a set of user-designed virtual components. These virtual components (like springs and dampers) pull the robot towards its goal and push it away from obstacles. The final trajectory emerges from these interactions, eliminating the need for separate path-planning computation [12], [13]. In contrast, traditional impedance control is primarily a motion control method. It assumes a pre-planned trajectory and makes the robot's end-effector behave compliantly along that path.

C. Multi-agent Collaboration

Multi-robot and human-robot collaboration both require the robot to avoid moving obstacles to manage physical contact. Previous works addressed collision avoidance between multiple robots in a shared workspace through motion coordination with trajectory replanning [25], [26] or reacting to repulsive actions from collision prevention regions, such as control barrier functions (CBFs), solved via quadratic programming [27]–[30]. Unlike the method proposed here, these solutions are often centralized or, even if implemented distributively, rely on the global exchange of information and variables among all agents.

Human collaboration with multi-robot systems have been addressed before [31], [32]. These works focused on physical collaboration rather than the safety of humans in shared workspaces and utilized centralized architectures. This motivated [33], [34] to develop a decentralized CBFs approach for safe human interactions with multiple robots in the same workspace. However, the method formulation only considered collision avoidance with the human without accounting for potential inter-robot collisions. Additionally, as a CBFs-based approach, it relied on availability of information from all agents. There remains a gap for treating all collaborative agents on the same level with minimal information exchange, enabling seamless interactions among all of them.

III. METHODOLOGY

A. Virtual Mechanism for Reach and Avoidance

The robot's reach and avoidance motions are generated by the interaction between its dynamics and a virtual mechanism. This mechanism consists of a goal spring that pulls the end-effector toward the next task waypoint, avoidance springs that repel other agents and obstacles, and associated dampers.

A spring and a damper attached between end effector at \mathbf{x}_{EE} and goal at \mathbf{x}_{goal} produces the force:

$$\mathbf{f}_{goal}(t) = k\mathbf{x}(t) + c\dot{\mathbf{x}}(t), \quad \mathbf{x}(t) = \mathbf{x}_{goal} - \mathbf{x}_{EE}(t) \quad (1)$$

at each time t , where k and c are the stiffness and damping coefficients. Gentle long-distance movements are obtained through filtering. Given a strictly monotone time law $s : [0, T] \subseteq \mathbb{R} \rightarrow [0, 1]$ such that $s(0) = 0$ and $s(T) = 1$, we take

$$\mathbf{x}(t) = \mathbf{x}_{EE}(0) + s(t)(\mathbf{x}_{goal} - \mathbf{x}_{EE}(0)) - \mathbf{x}_{EE}(t), \quad t \in [0, T] \quad (2)$$

For collision avoidance, we use springs with Gaussian energy profile $E(\mathbf{x}) = -k\sigma^2 \exp\left(-\frac{\|\mathbf{x}\|^2}{2\sigma^2}\right)$ which generate local repulsive force

$$\mathbf{f}_{avoid}(t) = k\mathbf{x}(t)\exp\left(-\frac{\|\mathbf{x}(t)\|^2}{2\sigma^2}\right), \mathbf{x}(t) = \mathbf{x}_{obj}(t) - \mathbf{x}_{EE}(t) \quad (3)$$

Here, denoting by f_{max} the maximum force parameter, $\sigma = f_{max}e^{0.5}/k$ is the distance from the obstacle \mathbf{x}_{obj} at which this maximum force occurs. Any two of k, f_{max}, σ uniquely determine the spring profile. Four examples of avoidance spring profile with varying f_{max} and σ are illustrated in Fig. 2. To implement inter-robot avoidance, twelve virtual points are distributed over each robot body, and avoidance springs are attached between every pair of points belonging to different robots. Additional avoidance springs are attached between the robot and other static obstacles, such as the table, with increased effective height when a block is grasped by the end-effector.

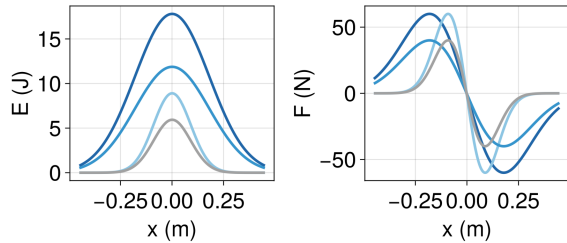


Fig. 2: Four avoidance spring profiles. Profile 1 (gray): $f_{max} = -40, \sigma = 0.09$; Profile 2 (light blue): $f_{max} = -60, \sigma = 0.09$; Profile 3 (medium blue): $f_{max} = -40, \sigma = 0.18$; Profile 4 (dark blue): $f_{max} = -60, \sigma = 0.18$.

For human hand avoidance, we also take advantage of unilateral saturating dampers. This element guarantees that a repulsive force is applied to the end effector (only) when a human hand is both (i) moving towards the end effector and (ii) within a specified proximity. Unlike potential field and control barrier functions which continuously enforce a minimum separation, the unilateral damper penalizes relative velocity. Any motion of the robot and hand towards each other is actively damped, allowing for proximity but only at low relative speed. If we denote the hand position by \mathbf{x}_{hand} , we have

$$\dot{r}(t) = \frac{\mathbf{d}(t)^T}{\|\mathbf{d}(t)\|} \dot{\mathbf{d}}(t) \quad \text{where} \quad \mathbf{d}(t) = \mathbf{x}_{hand}(t) - \mathbf{x}_{EE}(t). \quad (4a)$$

The damper is active only when $\dot{r} < 0$ (approaching motion); otherwise no force is applied. The force acting on the end effector is then given by

$$\mathbf{f}_{brake}(t) = \begin{cases} \min(-c(\|\mathbf{d}(t)\|)\dot{r}(t), f_{max}) \frac{\mathbf{d}(t)}{\|\mathbf{d}(t)\|}, & \dot{r} < 0, \\ 0, & \dot{r} \geq 0, \end{cases} \quad (4b)$$

where f_{max} is the maximum allowable magnitude of the avoidance force and the damping coefficient c is given by

$$c(\|\mathbf{d}(t)\|) = \begin{cases} c_0 \left(1 - \frac{\|\mathbf{d}(t)\|}{R}\right), & \|\mathbf{d}(t)\| < R, \\ 0, & \|\mathbf{d}(t)\| \geq R, \end{cases} \quad (4c)$$

The damper is only activated when the hand is within the activation radius R of the robot, and the damping coefficient c decays linearly with the distance $\|\mathbf{d}\|$ from its maximum value c_0 at contact to zero at R , ensuring a smooth transition.

To implement our virtual component-based controller, we used the VMRobotControl.jl package [35], a platform-independent framework designed for virtual model control in robotic systems. This package enables the construction and simulation of both the robot and virtual components such as masses, springs, and dampers. Given a robot's URDF file and virtual components' definition, the package computes the robot and virtual mechanism's dynamics.

B. Stall Detection and Conflict Resolution

Robot may face deadlock when the ensemble of forces acting on the robot results in a small equivalent force. This is a sign of conflicting requirements. To detect this deadlock, we consider the following metric

$$\rho = \|F_{net} - F_{tot}\|, \quad F_{net} = \left\| \sum_{i=1}^N \mathbf{f}_i \right\|, \quad F_{tot} = \sum_{i=1}^N \|\mathbf{f}_i\|, \quad (5)$$

where \mathbf{f}_i denotes the force generated by the i -th virtual component (goal or avoidance), and N is the number of components attached to the robot. When ρ exceeds a predefined threshold, the robot is considered in *stall* state, which leads to a negotiation procedure.

All robots in stall undergo a prioritization procedure, whose goal is to resolve the deadlock. The robot to prioritize is selected according to the following rules, which are considered only if no robot is currently prioritized. (i) Robots that have finished all tasks and are in rest position will not be prioritized. (ii) Robots executing a grasping or releasing motion are prioritized if they are within a proximity threshold of the closest robot. (iii) Robots that are very close to their target (within 6 cm) are given priority. If multiple robots satisfy (ii) or (iii), one is selected according to rule (iv). (iv) For all remaining robots, priority is assigned via a biased draw to guarantee fairness. Specifically, when more than two robots are stalled, one robot is prioritized according to the probability distribution:

$$\mathbb{P}(i) = \frac{\exp(-\alpha c_i)}{\sum_{j=1}^{n_{robot}} \exp(-\alpha c_j)}, \quad \alpha > 0, \quad (6)$$

where $c_i \in \mathbb{Z}$ is the number of times robot i has previously been prioritized and α is a parameter that controls how strongly past prioritizations reduce its future likelihood of selection. Once a robot is selected, its prioritization counter c_i is updated, and the procedure repeats next time robots are stalled. Biased draw in (iv) ensures fairness: robots with fewer prior selections are more likely to be chosen, balancing workload over time.

During prioritization, the selected robot proceeds directly toward its goal without avoiding other robots (still avoiding humans), while all other robots suspend goal-directed motion and only perform avoidance. After the prioritized robot

reaches its goal, all robots resume their original behaviors. In our experimental setting, all robots participate in prioritization since they operate in close proximity. In larger environments, however, prioritization can be restricted to local robots within a certain neighborhood. Note that (ii) is also applied when robots are not stalled to ensure safe and reliable block manipulation. While rare edge cases might still exist in coupling VMC with our prioritization procedure, deadlocks are always resolved in our experiments. A formal analysis is left to an extended journal version of this paper.

Conflict resolution is applied only to robots. Humans are naturally capable of assessing the situation and deciding an appropriate time to engage with the workspace, and therefore do not encounter deadlocks when collaborating with others. In contrast, robots with only the mechanical layer (VMC) lack such ability, they would remain stalled in deadlock situations. The conflict resolution layer thus provides robots with minimal decision-making to either take the lead or compromise. Human safety is always preserved by maintaining robot–human avoidance regardless of prioritization.

IV. IMPLEMENTATION AND EXPERIMENTAL PLATFORM

Two 6-DoF manipulators (UR5) are positioned on opposite sides of a table containing a 4×4 grid ($24\text{cm} \times 24\text{cm}$). Sixteen cubic blocks (3 cm) are distributed around the grid, as illustrated in Fig. 1. The task is to place all blocks into the grid cells. Placement proceeds from the center outward: robots first fill cells closer to the grid center; among cells at equal distance, selection is random. This ordering mimics an assembly process where central positions are completed before peripheral ones. Each robot initially selects blocks on its own side of the workspace and then proceeds to pick from the remaining blocks. For each pick-and-place action, task waypoints (\mathbf{x}_{goal}) are the block’s grasping position and the position above it.

An Intel RealSense D405 RGB-D camera tracks 21 human hand keypoints using MediaPipe [36] at 10 Hz for human avoidance. Avoidance performance of the robot is limited by the update rate of hand detection. Control and perception run on a standard PC.

The implementation of VMC on UR5s is based on a simulated virtual UR5 robots, using the VMRobotControl.jl package [35]. The interaction forces outlined in Section III-A drive the virtual UR5 robots. The simulated physics determines the motion and provide a reference configuration to each real UR5. Virtual springs and dampers for reaching and avoidance are connected (and/or disconnected during prioritization) between each virtual robot and relevant entities, such as task targets, physical robot’s position and the workspace boundaries. At each control cycle, we simulate the virtual robot’s dynamics under all virtual forces for a duration of Δt to obtain a joint-space target \mathbf{q}_d . This target is then sent to the physical robot. Stiffness and damping gains are selected experimentally to ensure sufficient repulsive force for reliable obstacle and human avoidance.

To assess scalability, we also conducted simulation tests with up to four robots. The setup is shown in Fig. 3.

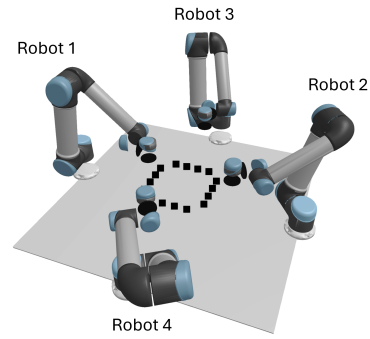


Fig. 3: Multi-robot simulation. Robot 1 and 2 are for two-robot experiment. Robot 1-3 are for three-robot experiment.

V. EXPERIMENT AND RESULTS

A. Agent-agnostic Collaboration

We demonstrate the collaboration between a robot and either another robot or a human. In this experiment, Robot 1 tries to maintain its position in the center of the grid, while the other agent (robot or human) places all blocks. Robot reaching is implemented through attractive springs. Robot 1 avoids Robot 2 or the human via avoidance springs. In contrast, Robot 2 adopts an aggressive behavior and does not avoid Robot 1. No communication or any form of negotiation is implemented between the agents. Robot 1 only perceives the position of human hand and the other robot. The parameters of the virtual mechanisms are in Table II.

Fig. 4 presents sample trajectories from these experiments. Color indicates time, with matching colors identifying synchronous events. No deadlock occurred. Potential conflicts were resolved at the mechanical level, taking advantage of the different characteristics of each robot, due to the tuning of their control parameters. Fig. 4 shows that Robot 1 successfully avoided both Robot 2 and human hand while using the same virtual mechanism, demonstrating agent-agnostic behavior.

To quantify spatial clearance, we define the minimum robot–robot distance d_{min}^{RR} as the smallest Euclidean distance between the end-effectors of two robots, and the minimum robot–human distance d_{min}^{RH} as the smallest distance between a robot end-effector and the closest tracked point of the human hand. The minimum inter-robot distance was $d_{min}^{RR} = 16.6\text{cm}$, and the minimum robot–human distance averaged over five trials with different participants was $d_{min}^{RH} = 21.8\text{cm}$, confirming that safe separation was consistently maintained during interaction.

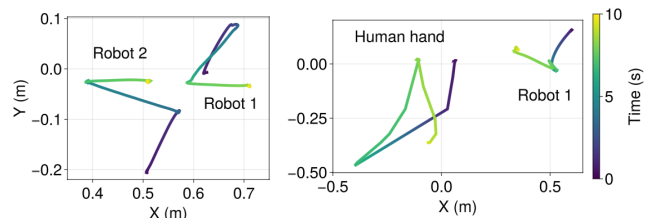


Fig. 4: Robot 1 avoiding Robot 2 (left) and a human (right) with the same VMC avoidance spring. In both cases, Robot 1 retreats as the other agent approaches, then returns.

B. Safety-aware Collaboration

We investigate how different spring profiles influence the robot behavior. The task is the same as the last section, within the setting of Fig. 1. Four avoidance spring profiles are illustrated in Fig. 2. Increasing σ extends the range of repulsive force, while a larger f_{\max} increases the force magnitude. We ran five trials with different participants for each spring profile. Participants rested and re-entered the scene at least four times to encourage interaction with the robot. The resulting minimum distances d_{\min}^{RH} are in Table I.

We also examined Speed and Separation Monitoring (SSM) requirements, as specified in ISO/TS 15066 [37]. To illustrate how parameter tuning can be used to meet SSM requirements, we computed a conservative protective separation distance S_p given by

$$S_p = S_h + S_r + S_s + C + Z_d + Z_r, \quad (7)$$

where $S_h = v_h(T_r + T_s)$ accounts for human motion, $S_r = v_r T_r$ for robot reaction time, S_s for robot stopping distance, C for intrusion distance, and Z_d and Z_r for uncertainty margins on human sensing and robot pose. Considering the size of the working space, we assumed max human speed in the direction of robot $v_h = 0.8 \text{ m s}^{-1}$, max robot speed in the direction of operator $v_r = 0.3 \text{ m s}^{-1}$, controller reaction $T_r = 0.08 \text{ s}$, mid-speed stopping time $T_s = 0.15 \text{ s}$ (UR5 datasheet), $C = 0.08 \text{ m}$ for hand-resolution sensing, and sensor / pose uncertainties $Z_d = Z_r = 0.02 \text{ m}$. Taking $S_s \simeq \frac{1}{2} v_r T_s$ yields $S_p \simeq 0.35 \text{ m}$. Table I summarizes the total amount of time $T_{<S_p}$ the robot does not respect the separation distance S_p (averaged over experiments, each lasting above one minute).

Increasing either σ or f_{\max} alone (profiles 1–3) does not substantially affect d_{\min}^{RH} , whereas increasing both (profile 4) yields a larger d_{\min}^{RH} . This effect is more pronounced for participants who moved cautiously, while for participants who moved faster the improvement was smaller, leading to larger variance in profile 4. $T_{<S_p}$ decreases as f_{\max} increases, and further when σ increases, reaching a minimum when both parameters increased. This highlights that SSM compliance can be influenced directly by control parameter settings.

Finally, we examined the effect of the damping mechanism described in Section III-A. Five experiments with different participants were performed using spring (profile 1) and damper ($c_0 = 150 \text{ N s m}^{-1}$, $R = 0.5 \text{ m}$, $f_{\max} = 50 \text{ N}$). The damper reacts proportionally to the human hand’s velocity, thereby providing stronger repulsion when the hand approaches the robot at high speed and reducing the robot’s motion toward the hand. This reduced both d_{\min}^{RH} and $T_{<S_p}$ compared to the spring-only baseline (profile 1). Responsiveness would likely be improved with higher-rate perception, currently limited at 10 Hz.

C. Conflict resolution

While robots with non-conflicting tasks resolve interaction mechanically as shown in Section V-A, in general robots sharing a common workspace may stall. This typically occurs when two robots have incompatible tasks and symmetric

TABLE I: Minimum separation d_{\min}^{RH} and violation duration $T_{<S_p}$ under different avoidance profiles and mechanisms.

Spring profile	d_{\min} (cm)	$T_{<S_p}$ (s)
Spring profile 1 (baseline)	21.8 ± 2.0	4.6 ± 1.8
Spring profile 2 (increased f_{\max})	24.4 ± 4.7	2.3 ± 1.1
Spring profile 3 (increased σ)	24.8 ± 9.1	0.9 ± 0.9
Spring profile 4 (both increased)	30.0 ± 9.8	0.5 ± 0.5
Spring profile 1 and damper	19.9 ± 2.9	4.1 ± 0.5

policies to realize them. We examine such cases and show how the prioritization strategy of Section III-B resolves them.

Consider the scenario in Fig. 5, where two robots transport blocks along intersecting straight-line trajectories. Without negotiation, the attractive and avoidance forces balance near the intersection, resulting in a stall. This is illustrated through time-trajectories in Fig. 6 (top-left). The motion of the robots stalls right after 25s, and the two-block task is still not completed after 150 s. With our prioritization strategy, stalls are detected via the force-based metric (5) with the threshold for ρ set to 4.0 N. A priority is negotiated, and spring stiffnesses are adjusted accordingly. This allows the robots to resolve the deadlock and complete the task successfully, as shown in Fig. 6 (bottom).

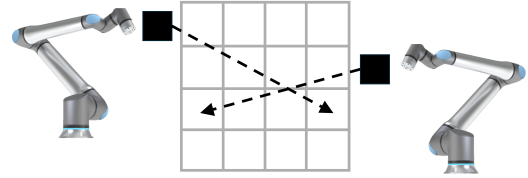


Fig. 5: Two UR5 robots transport cubic blocks (black squares) toward their target cells (dashed arrows). The intersection of trajectories represents the potential conflict region.

To analyze our conflict resolution strategy, we also consider a scenario in which two robots each take one of the 16 blocks and move it to one of the 4×4 inner grid along straight-line trajectories. A combinatorial enumeration with geometric intersection checks shows that, under random choices, the probability of two trajectories intersecting is 28.4%. With three robots in the scene, the possibility increases up to 61.2 % and with four robots it becomes even higher. Assuming that every intersection leads to a deadlock, conflicts would occur in nearly two thirds of tasks. With the proposed stall detection and negotiation, we observed no unresolved deadlocks across all runs in the evaluated settings.

We further examined how control parameters influence task completion time, which is measured with elapsed time until the last block is placed. In this experiment, two robots placed all blocks into the grid as described in Section IV, without human involvement. Three sets of ‘Slow’, ‘Medium’ and ‘Fast’ parameters summarized in Table IIb yielded three distinct completion times, with five trials conducted for each case: 144.5 ± 6.9 , 103.7 ± 4.5 and $88.5 \pm 4.8 \text{ s}$. Faster completion was achieved with stiffer springs connected to the goal with larger maximum forces and higher speeds in the moving-goal filter.

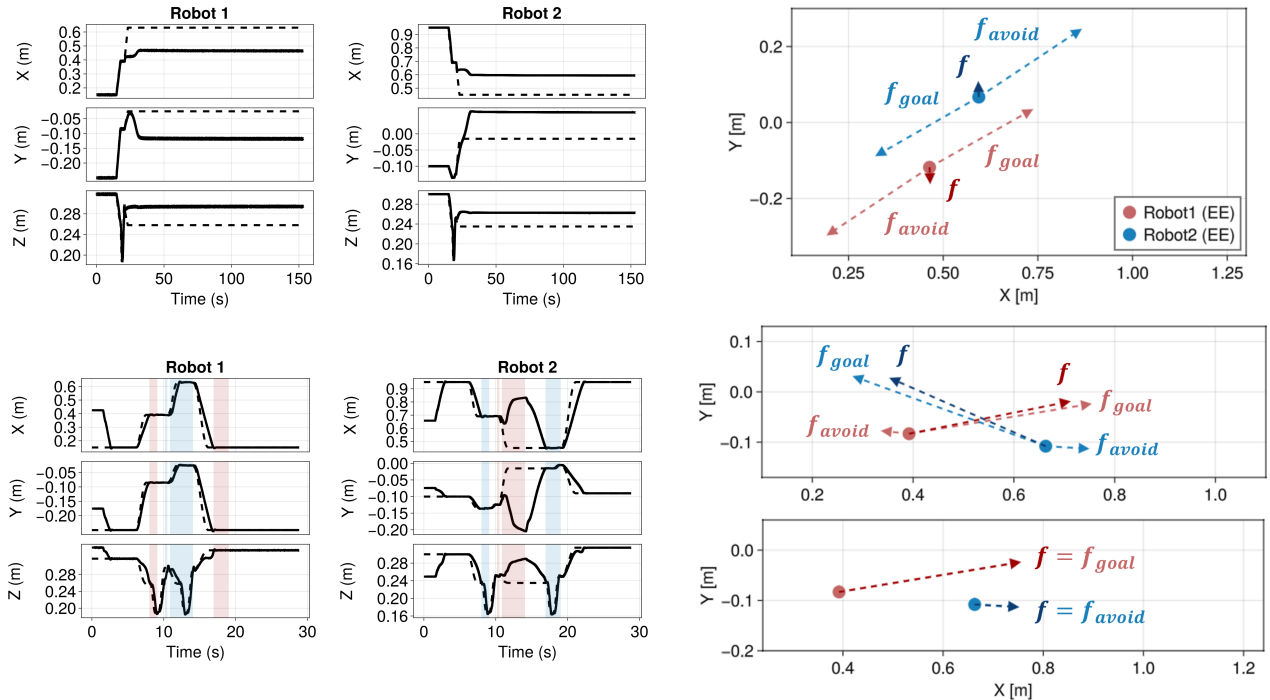


Fig. 6: Top: without conflict resolution layer, robots entered a deadlock due to force balance at $t = 25$ s and did not recover. Bottom: with the conflict resolution, robots resolved the conflict by prioritizing robot 1 at 10 s. Top-Left: robot trajectories (dashed - desired EE position; solid - EE position). Top-Right: balancing forces at $t = 150$ s, indicating a stall; f_{goal} goal-reaching force, f_{avoid} cumulative avoidance force, $f = f_{goal} + f_{avoid}$. Bottom-Left: robot trajectories. Blue - prioritized interval, Red - non-prioritized interval. Bottom-Right: top - stall detected at $t = 10.87$ s, bottom - stall resolved at $t = 10.95$ s

D. Scalability

Finally, we assess the the ability of our virtual model control algorithm to handle robot team of different sizes. We consider two main scenarios. **Scenario A**: all blocks are placed into grid according to Section IV. **Scenario B** each robot places blocks only within its own half of the grid (reduced level of interaction). For each scenario and each team size, five experiments/simulations were conducted.

Simulations with three and four robots were conducted with the robots placed as shown in Fig. 3. In Scenario B with more than two robots, each robot was assigned a predefined range of blocks closer to its workspace. Since real-world experiment time is influenced by hardware limitations, the completion time in real-world experiment is not directly comparable to completion time in simulation. For consistency, we also report simulation results for the one- and two-robot cases.

To characterize interaction complexity, for every pair of simultaneous block transfers, we test if their ideal straight-line source-to-destination segments intersect (crossing event) or get very close (within 0.1 m, near-miss event). We report the *total crossing time* T_{cross} and *near-miss time* T_{nm} . Prioritization frequency counts how often each robot is selected during negotiation, revealing potential asymmetries in allocation.

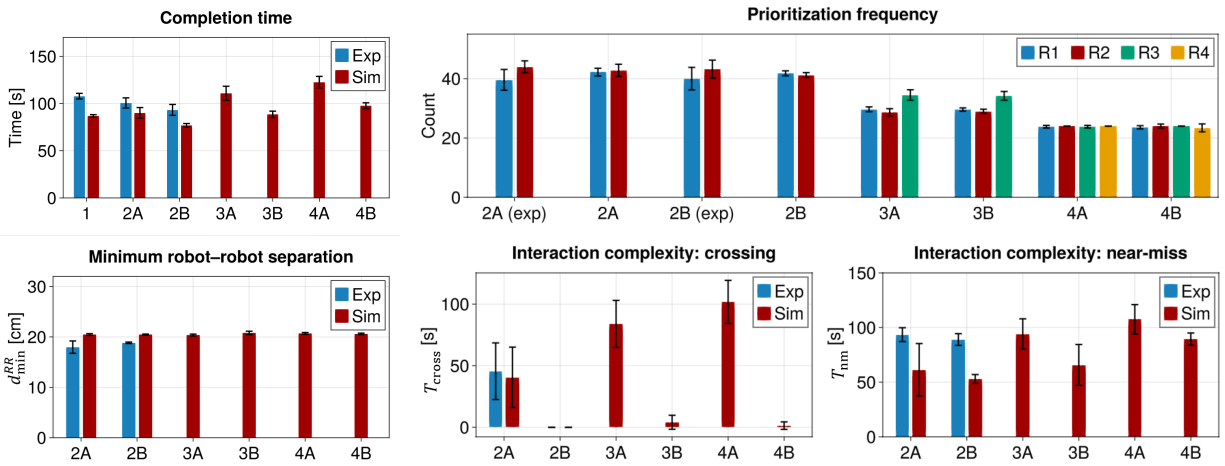
Results are summarized in Fig. 7a. T_{cross} and T_{nm} show that Scenario A involves more interaction between robots than Scenario B. Both experiments and simulations show that the

completion time improves up to two agents with reduced interaction (Scenario B); beyond this point, the limited workspace relative to robot size increases the completion time. Nevertheless, even in complex scenarios with four robots sharing the same workspace, all tasks were successfully completed without deadlocks. Prioritization counts were evenly distributed across robots, indicating fair workload allocation. Across all settings, robots successfully avoided collisions, maintaining an average separation of about 20 cm regardless of team size.

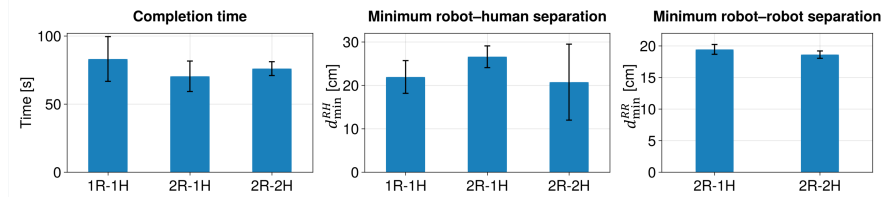
For mixed human–robot collaboration, the grid was partitioned into a checkerboard pattern: cells of one color were assigned to humans for block placement, while the remaining cells were reserved for robots. This ensures sufficient human–robot interactions. Scenario A is used when two robots were present. For each team size, five experiments were conducted with different participants.

Task was successfully completed with different team size while maintaining around 20 cm between robots and robot-human as shown in Fig. 7b. Completion time improved slightly when increasing from two to three agents, but no further gains were observed with larger teams.

Our decentralized approach enabled seamless adaptation to different numbers of agents without altering the control structure: outside of conflict resolution, each robot relies only on the position of other agents which is obtainable with onboard camera. During the experiments, humans occasionally rested by withdrawing from the workspace and later re-entered



(a) **Multi-robot (real-world experiments and simulations).** Performance vs. team size. X-axis denotes number of robots and scenario. R_n denotes each robot in the simulation.



(b) **Multi-agent (robots and humans).** Performance metrics vs. team size. $nR-mH$ denotes collaboration between n robots and m humans.

Fig. 7: Completion time, safety and interaction metrics across scenarios and team sizes.

to continue the task. The proposed framework naturally accommodates such changes, allowing collaboration despite dynamic team composition.

VI. CONCLUSIONS AND FUTURE WORK

This paper presented a decentralized, agent-agnostic, safety-aware framework for HRC based on VMC. Humans and robots share a common workspace. Motion planning and motion control are blended through virtual model control, enabling fast/slow reactive robot behavior based on intuitive parameter tuning. Deadlocks are detected through a force-balancing condition and resolved via a prioritization procedure that adapts the virtual model controller, reducing the probability of deadlocks. The method is highly scalable, with practical limits determined mainly by the computation of virtual forces and communication of sensor information. The approach has been validated experimentally on position-controlled UR5 manipulators with up to two robots and two humans, and in simulation with up to four robots while maintaining inter-agent separation at around 20 cm, demonstrating (i) unified treatment of humans and robots, (ii) experimental deadlock-free operation, and (iii) scalability to larger teams.

In this paper, we focus on the distributed control architecture and therefore do not investigate how humans adapt their behavior in response (e.g., whether they become more assertive or more hesitant to enter the scene). A dedicated user study is an important direction for future work to evaluate how VMC shapes the collaborative workspace from the human perspective (usability, comfort, acceptance, etc.).

Integration of human feedback within an adaptive virtual model control setting is also an important research direction. This could take into account non-verbal cues such as facial expressions or gestures to improve comfort, safety, and trust during human-robot interaction. The physical interpretability of VMC makes this adaptation tractable, offering a practical way toward user-centered collaboration without full redesign of the underlying controller.

REFERENCES

- [1] M. Vasic and A. Billard, "Safety issues in human-robot interactions," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 197–204.
- [2] Z. Saleem, F. Gustafsson, E. Furey, M. McAfee, and S. Huq, "A review of external sensors for human detection in a human robot collaborative environment," *Journal of Intelligent Manufacturing*, vol. 36, no. 4, pp. 2255–2279, 2025.
- [3] G. Hoffman, T. Bhattacharjee, and S. Nikolaidis, "Inferring human intent and predicting human action in human-robot collaboration," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 7, 2024.
- [4] W. Li, Y. Hu, Y. Zhou, and D. T. Pham, "Safe human-robot collaboration for industrial settings: A survey," *Journal of Intelligent Manufacturing*, vol. 35, no. 5, pp. 2235–2261, Jun. 2024.
- [5] D. Mukherjee, K. Gupta, L. H. Chang, and H. Najjaran, "A Survey of Robot Learning Strategies for Human-Robot Collaboration in Industrial Settings," *Robotics and Computer-Integrated Manufacturing*, vol. 73, p. 102231, Feb. 2022.
- [6] C. Fan, K.-F. Chu, X. Wang, K.-W. Kwok, and F. Iida, "State transition learning with limited data for safe control of switched nonlinear systems," *Neural Networks*, vol. 180, p. 106695, 2024.
- [7] F. Sanfilippo, M. H. Zafar, T. Wiley, and F. Zambetta, "From caged robots to high-fives in robotics: Exploring the paradigm shift from human-robot interaction to human-robot teaming in human-machine interfaces," *Journal of Manufacturing Systems*, vol. 78, pp. 1–25, 2025.

TABLE II: Control parameters for each experiment.

(a) Agent-agnostic and safety-aware collaboration (Section V-A and V-B).

Goal spring		Robot 1 Avoidance spring (for other robots)		Avoidance spring (for human hand)		Robot 2 Goal spring	
Stiffness [N/m]	f_{max} [N]	Stiffness [N/m]	f_{max} [N]	σ [m]	f_{max} [N]	Stiffness [N/m]	f_{max} [N]
1000	10	-1000	-40	0.18	-60	3000	20

(b) Conflict resolution (Section V-C). All robots have the same spring profiles. ‘Medium’ is used for showcase of conflict resolution.

	Goal spring		Filter goal	Avoidance spring (for other robots)	
	Stiffness [N/m]	f_{max} [N]	Speed [m/s]	Stiffness [N/m]	f_{max} [N]
Slow	2000	15	0.1	-900	-40
Medium	2000	20	0.4	-900	-40
Fast	3000	30	1.0	-900	-40

(c) Scalability (Section V-D). All robots have the same spring profiles.

Exp	Goal spring		Filter Goal	Avoidance spring (for other robots)		Avoidance spring (for human hand)	
	Stiffness [N/m]	f_{max} [N]	Speed [m/s]	Stiffness [N/m]	f_{max} [N]	σ [m]	f_{max} [N]
3	2000	20	0.4	-900	-40	0.18	-60

- [8] L. Mingyue Ma, T. Fong, M. J. Micire, Y. K. Kim, and K. Feigh, “Human-robot teaming: Concepts and components for design,” in *Field and service robotics: results of the 11th international conference*. Springer, 2017, pp. 649–663.
- [9] J. E. Pratt, C.-M. Chew, A. Torres, P. Dilworth, and G. A. Pratt, “Virtual model control: An intuitive approach for bipedal locomotion,” *The Int. Journal of Robotics Research*, vol. 20, pp. 129 – 143, 2001.
- [10] G. Chen, S. Guo, B. Hou, and J. Wang, “Virtual model control for quadruped robots,” *IEEE Access*, vol. 8, pp. 140 736–140 751, 2020.
- [11] A. Winkler, I. Havoutis, S. Bazeille, J. Ortiz, M. Focchi, R. Dillmann, D. Caldwell, and C. Semini, “Path planning with force-based foothold adaptation and virtual model control for torque controlled quadruped robots,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 6476–6482.
- [12] Y. Zhang, D. Larby, F. Iida, and F. Forni, “Virtual model control for compliant reaching under uncertainties,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 795–801.
- [13] D. Larby and F. Forni, “Optimal virtual model control for robotics: Design and tuning of passivity-based controllers,” *arXiv preprint arXiv:2411.06627*, 2025.
- [14] Y. Zhang, F. Iida, and F. Forni, “Periodic robust robotic rock chop via virtual model control,” *arXiv preprint arXiv:2508.02604*, 2025.
- [15] D. Larby, J. Kershaw, M. Allen, and F. Forni, “Collaborative drill alignment in surgical robotics,” *arXiv preprint arXiv:2503.05791*, 2025.
- [16] J. Lyu, P. Ruppel, N. Hendrich, S. Li, M. Görner, and J. Zhang, “Efficient and Collision-Free Human–Robot Collaboration Based on Intention and Trajectory Prediction,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 15, no. 4, pp. 1853–1863, Dec. 2023.
- [17] A. Pupa, M. Minelli, and C. Secchi, “A Dynamic Planner for Safe and Predictable Human-Robot Collaboration,” *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 507–514, Jan. 2024.
- [18] K. Merckaert, B. Convens, M. M. Nicotra, and B. Vanderborght, “Real-time constraint-based planning and control of robotic manipulators for safe human–robot collaboration,” *Robotics and Computer-Integrated Manufacturing*, vol. 87, p. 102711, Jun. 2024.
- [19] C. Escobedo, M. Strong, M. West, A. Aramburu, and A. Roncone, “Contact anticipation for physical human–robot interaction with robotic manipulators using onboard proximity sensors,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 7255–7262.
- [20] F. Flacco, T. Kröger, A. De Luca, and O. Khatib, “A depth space approach to human-robot collision avoidance,” in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 338–345.
- [21] N. Hogan, “Impedance control of industrial robots,” *Robotics and computer-integrated manufacturing*, vol. 1, no. 1, pp. 97–113, 1984.
- [22] L. Bertoni, L. Baccelliere, L. Muratore, and N. G. Tsagararakis, “A Proximity-Based Framework for Human-Robot Seamless Close Interactions,” *IEEE Robotics and Automation Letters*, vol. 10, no. 8, pp. 8514–8521, Aug. 2025.
- [23] S. Ktistakis, L. Gimeno, F.-Z. Laftissi, A. Hoss, A. De Donno, and M. Meboldt, “Robot assistance primitives with force-field guidance for shared task collaboration,” *Robotics and Computer-Integrated Manufacturing*, vol. 96, p. 103061, 2025.
- [24] M. Lagomarsino, M. Lorenzini, E. De Momi, and A. Ajoudani, “PRO-MIND: Proximity and Reactivity Optimization of Robot Motion to Tune Safety Limits, Human Stress, and Productivity in Industrial Settings,” *IEEE Transactions on Robotics*, vol. 41, pp. 2067–2085, 2025.
- [25] J. K. Behrens, K. Stepanova, and R. Babuska, “Simultaneous task allocation and motion scheduling for complex tasks executed by multiple robots,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 11 443–11 449.
- [26] S. Zhang and F. Pecora, “Online and scalable motion coordination for multiple robot manipulators in shared workspaces,” *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 3, pp. 2657–2676, 2023.
- [27] D. Sun and Q. Liao, “Real-time coordination of multiple robotic arms with reactive trajectory modulation,” *IEEE Transactions on Robotics*, 2024.
- [28] L. Lindemann and D. V. Dimarogonas, “Control barrier functions for multi-agent systems under conflicting local signal temporal logic tasks,” *IEEE Control Systems Letters*, vol. 3, no. 3, pp. 757–762, 2019.
- [29] B. Capelli and L. Sabattini, “Connectivity maintenance: Global and optimized approach through control barrier functions,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 5590–5596.
- [30] X. Tan and D. V. Dimarogonas, “Distributed implementation of control barrier functions for multi-agent systems,” *IEEE Control Systems Letters*, vol. 6, pp. 1879–1884, 2022.
- [31] N. Dehio, J. Smith, D. L. Wigand, P. Mohammadi, M. Mistry, and J. J. Steil, “Enabling impedance-based physical human–multi–robot collaboration: Experiments with four torque-controlled manipulators,” *The International Journal of Robotics Research*, vol. 41, no. 1, pp. 68–84, Jan. 2022.
- [32] R. Wen, Q. Rouxel, M. Mistry, Z. Li, and C. Tiseo, “Collaborative bimanual manipulation using optimal motion adaptation and interaction control: Retargeting human commands to feasible robot control references,” *IEEE Robotics & Automation Magazine*, vol. 31, no. 4, pp. 68–80, 2023.
- [33] K. Shi, J. Chang, S. Feng, Y. Fan, Z. Wei, and G. Hu, “Safe human dual-robot interaction based on control barrier functions and cooperation functions,” *IEEE Robotics and Automation Letters*, 2024.
- [34] K. Shi, J. Liu, J. Chang, and G. Hu, “Distributed quadratic programming for safe human multirobot interaction under task constraints,” *IEEE/ASME Transactions on Mechatronics*, 2025.
- [35] D. Larby, “VMRobotControl.jl,” <https://github.com/Cambridge-Control-Lab/VMRobotControl>, 2024.
- [36] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, “Mediapipe hands: On-device real-time hand tracking,” *arXiv preprint arXiv:2006.10214*, 2020.
- [37] “Iso/ts 15066:2016 – robots and robotic devices – collaborative robots,” *International Organization for Standardization*, 2016.