Collective Transport of Unconstrained Objects via Implicit Coordination and Adaptive Compliance

Nicole E. Carey*†‡
nic.carey@autodesk.com

Justin Werfel*†
jkwerfel@seas.harvard.edu

*Wyss Institute for Biologically Inspired Engineering, Harvard University, Boston, MA 02115

†John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02139

‡Autodesk Robotics Laboratory, San Francisco, CA 94111

***Abstract*—We present a decentralized control algorithm for robots to aid in carrying an unknown load. Coordination occurs solely through sensing of the forces on or movement of the shared load. Robots prevent undesired motion of the load while permitting movement in the task-relevant subspace, and stabilize against unexpected events by a transient decrease in compliance. The algorithm requires no direct communication between agents, and minimal knowledge of the system or task. We demonstrate the approach in simulation using a commercially available compliant robotic platform.**

I. Introduction

Collective transport is the phenomenon or task in which an arbitrary number of independent agents move an object too heavy or unwieldy for one to handle alone. In the natural world, this activity is most commonly associated with ants [1], [2]; in swarm robotics, it is a mechanism allowing groups of small robots to move much larger items [3], [4]. A sub-category of this task might be termed "collaborative transport": one informed agent has a destination in mind, and the others aid in moving an object without necessarily having knowledge of the goal or environment. Imagine needing to move a large table: one person may have an intended target location, and recruit a group of friends, who lift the table and then follow the first person's lead in moving it in the horizontal plane. In such a scenario, the coordination between carriers may take place without explicit communication; each follower can feel the motion of the table and respond accordingly.

In this work we present a decentralized control scheme that allows independent autonomous robots to act as the followers in this scenario. No direct communication is required, nor knowledge of the number of other robots or geometric or physical properties of the object being carried. The approach can accommodate physical perturbations and changes in the number of robots mid-task, and does not require specialized gripping hardware that constrains the available degrees of freedom to conform to theoretical assumptions.

This work was supported by a Space Technology Research Institutes grant (number 80NSSC19K1076) from NASA's Space Technology Research Grants Program.

The approach involves decoupling the object's motion in the horizontal plane from movement out of plane; each robot stabilizes the object against movement corresponding to roll and pitch, while permitting and following horizontal movement, which it uses as a proxy to gauge forces exerted on the object by the leader. A key element of the approach is to use active changes in compliance to respond to sudden movements of the object. This strategy corresponds to that intuitively used by the people in the table-moving analogy: in response to an unexpected shift, a human temporarily stiffens their muscles. Similarly, an adaptive stiffness term in the robot control law stabilizes against large system unknowns or disturbances.

The approach is generalizable to any robot platform with sufficient degrees of freedom and actively compliant joint control. Here we present the control algorithm and its theoretical basis, and demonstrate its operation in simulation using a commercially available mobile manipulator.

II. Related Work

Much of the past work in decentralised collective transport using force as an implicit communication medium is constrained to two dimensions (i.e., pushing or pulling an object along a flat surface) [5], [6], and often requires explicit communication between agents [7] or a shared knowledge base encompassing trajectory and/or payload information [8], [9]. Previous work on shared manipulation in three dimensions requires knowledge of the inertial parameters of the payload [10], [11]; these can be determined at run-time by the robots, but doing so involves an extensive parameter estimation phase, in which robot behavior is explicitly coordinated and the number of agents is known beforehand [11].

When moving from theory or simulation to physical reality, the control constraints of many collective transport algorithms frequently assume restrictions on motion that then require specialized gripper hardware to satisfy [12]–[14]. This need to limit the degrees of freedom in the interaction, and the accompanying necessity for bespoke hardware or specialised payload affordances, can restrict the ability of

robots using these approaches to spontaneously transport objects, or to handle widely varying load types.

III. Challenges and approach

For a single robot seeking to transport an unknown object, lack of information about the inertial properties of the load is one of the main impediments to developing stable and effective force-feedback control. Without an accurate full-system model, calculating the appropriate joint control commands to counteract internal and interaction wrenches on the payload is challenging, especially if the load is heavy and/or has a mass centroid far from the grasp point. Such a scenario can easily lead to system instability. For a group of collaborative robots without the ability to directly coordinate, the issue is compounded by disturbances from the motion of other agents which result in additional external forces transmitted through the payload.

Assuming the requirements of a transportation task are largely concerned with in-plane motion, we can expect most out-of-plane torques and vertical forces to be due to the mass and inertial properties of the load to be transported. Hence, we can decompose the collective follower's task into two parts, firstly the identification of planar forces that indicate a guiding impetus, and secondly the rejection of out-of-plane torques and forces. We must accomplish the latter without generating conflicts with guiding forces or the other agents' attempts at stabilisation of the same object. In essence, the robots must be able to settle on a mutually agreed force equilibrium at the end-effector. We can enable this by allowing some degree of compliance in both the end-effector and joint space of each robotic agent (Fig. 1), and tuning this compliance according to the priorities of the collaborative task. Hence, we must fully decouple each robot's impedance behaviour in the end-effector space, and project these force and compliance constraints to torque and stiffness commands in the joint space.

Our approach implements an adaptive joint torque controller on each robot that uses only local sensor information. A human carrying an unknown load tends to adjust their joint stiffness according to the weight and/or out-of-plane rotation of the load [15]. Similarly, we introduce a variable stiffness control in the joint space to compensate for errors in the task space (estimated by a disturbance observer) which cannot be compensated for by the internal system model of each robot (e.g., a payload with significant weight or off-axis inertia).

Finally, in order to generate in-plane signals that can be used for reaching a directional consensus during transport, we decrease each robot's end-effector stiffness in the X-Y plane (as estimated from the robot's local frame of reference). The follower robots can obtain the needed information from the load either by sensing forces exerted on the gripper, or by registering movement of the load via that of the end-effector. An advantage of the latter approach is that the signal from planar motion is minimally affected by forces exerted in other dimensions, and can be used to robustly detect the in-plane forces exerted by a leader. These signals inform an outer control loop (acting independently of the robot's manipulator linkages), which then drives the mobile base of the robot along the environmental plane.

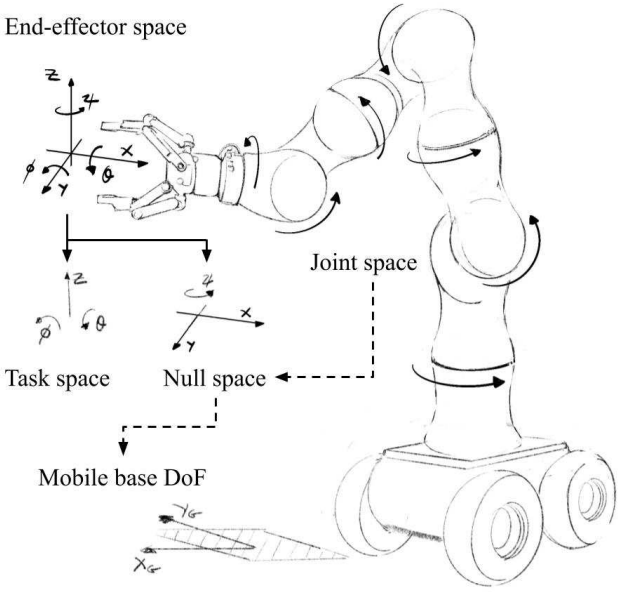

Fig. 1. System configuration and degrees of freedom. The end-effector space comprises the 6DoF pose in the world frame. The joint space of the robot comprises the rotational degrees of freedom of the manipulator (7 in this example). We can decompose the end-effector space into a high-accuracy "task space" and a low-priority "null space", which also includes the joint positions. Null-space control forces are de-prioritized until the desired task space accuracy has been achieved. Information from the end-effector can be used to control the motion of the mobile base.

IV. Control methodology

A. Core assumptions

Each robot agent is assumed to have a full suite of joint state sensors (position/velocity/acceleration), and a moderately accurate internal model of its own dynamic and kinematic parameters. For this paper, it is also assumed that the robots are equipped with the necessary sensors and analytical subsystems to negotiate a secure grasp on the payload object. Finally, we assume that the transportation task involves only lateral motion with no intentional out-of-plane rotation imparted by the lead agent, and that the object to be transported is rigid and can be grasped securely without significant slippage.

TABLE I
Key control variables

Variable Description	Notation
Position of end-effector in robot base frame	(x, y, z)
Orientation of end-effector in robot base frame	(θ, ϕ, ψ)
Joint state vector (angular position/velocity/acceleration)	$(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$
Task space impedance gains	$\mathbf{B}_e, \mathbf{K}_e$
Null space impedance gains	$\mathbf{B}_\nu, \mathbf{K}_\nu$
Control gains for mobile robot base	B_m, K_m

B. Object stabilisation

We consider each individual robot agent to consist of a multi-DoF manipulator mounted on a mobile base. To fully

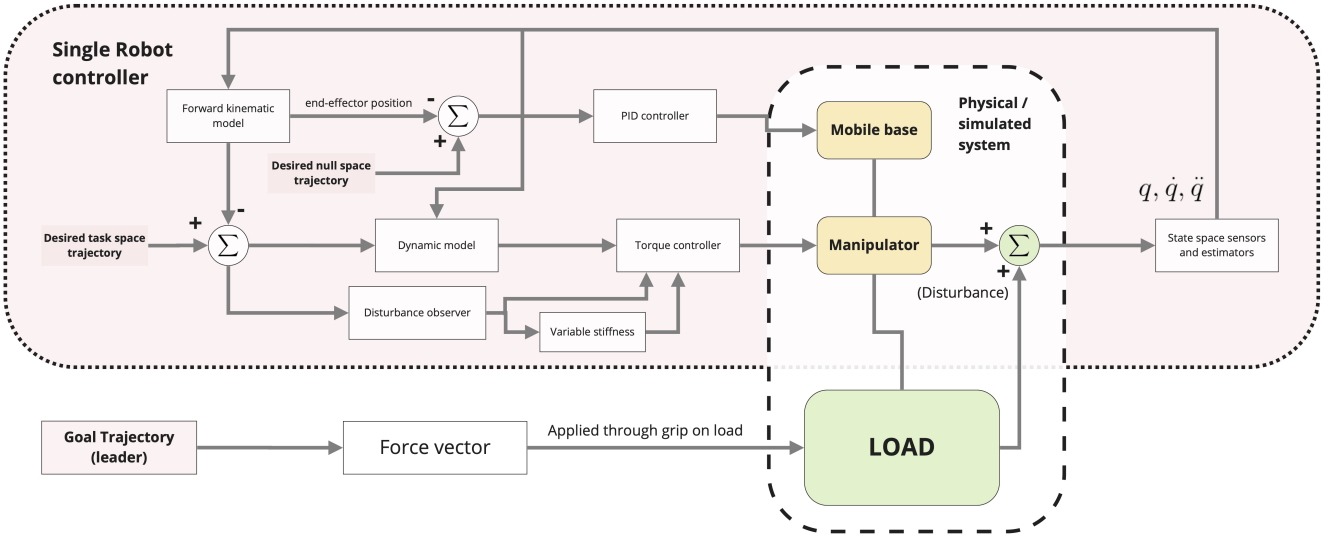

Fig. 2. A diagrammatical representation of the control system for each independent agent. Internal sensing enables accurate forward and inverse kinematic and (unloaded) dynamic models of each robot's manipulator. The mobile base control loop uses only position information fed to the system from the manipulator. The leader (another independent agent) can apply a guiding force at any point on the load. Neither leader nor followers have any knowledge of the number or location of the other collaborative agents.

control the task space impedance of the end effector, we need to map the forces experienced in this task space to the corresponding joint space torques, and differentiate the torques resulting from forces we wish to reject (e.g., gravity, disturbance forces) from those the robot should follow (e.g., forces applied by a leader). We adapt a method described in [16] to decouple the force control of a redundant manipulator by devolving the control law into task space vs. null space elements, where the task space describes the end-effector motion where high accuracy is desired, and the null space includes all remaining degrees of freedom. This type of control is commonly used to absorb collisions via the compliant joint space without affecting task space performance, thus ensuring robot safety around human operators.

Assuming the transport task consists largely of motion in the plane, and out-of-plane wrenches are likely to be from payload inertia or external disturbances, we choose our task space to comprise the vertical displacement and out-of-plane rotation of the payload (z, θ, ϕ), while the null space consists of the robot's n-dimensional joint space $(\mathbf{q})$ and the end-effector motion in the lateral world plane (x, y, ψ).

The decoupled control algorithm is formulated as follows. The manipulator and the portion of the payload it grasps can be considered as an independent subsystem, with the following local system dynamics:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_{ext} = \boldsymbol{\tau} \tag{1}$$

where $\boldsymbol{\tau}_{ext}$ is the $(n \times 1)$ vector of external torques resulting from interaction with the environment (and which are likely to include the distributed payload inertia and any forces imparted by the actions of other robot agents). While these torques can be directly measured via appropriately placed sensors, we can also design a disturbance observer using other signals from the workspace (Figure 2).

We can write a generalised impedance control law of the following form:

$$\mathbf{M}_d(\ddot{\mathbf{q}}_d - \ddot{\mathbf{q}}) + \mathbf{B}_d(\dot{\mathbf{q}}_d - \dot{\mathbf{q}}) + \mathbf{K}_d(\mathbf{q}_d - \mathbf{q}) = \boldsymbol{\tau}_{ext} \tag{2}$$

where $\mathbf{M}_d, \mathbf{B}_d, \mathbf{K}_d$ describe the desired impedance behaviour of the end-effector. Let $\mathbf{M}_d = \mathbf{M}$; then the desired impedance behaviour can be achieved via the following joint torque control:

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}_c + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_{ext} \tag{3}$$

where $\ddot{\mathbf{q}}_c$ is the command joint acceleration

$$\ddot{\mathbf{q}}_c = \ddot{\mathbf{q}}_d + \mathbf{M}^{-1}(\mathbf{B}_d\dot{\tilde{\mathbf{q}}} + \mathbf{K}_d\tilde{\mathbf{q}} - \boldsymbol{\tau}_{ext}) \tag{4}$$

and $\tilde{\mathbf{q}} = \mathbf{q}_d - \mathbf{q}$ is the joint space error.

Using this control formulation, we decompose the command torque into task-space and null-space elements:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{task} + \boldsymbol{\tau}_{null} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) \tag{5}$$

where $\boldsymbol{\tau}_{task}$ is the joint space torque command that satisfies the task space motion requirements, and $\boldsymbol{\tau}_{null}$ is the torque controlling the impedance behaviour of the remaining (null) degrees of freedom, including the joint impedance.

1) Task space control torque: We choose a task space acceleration control of the form:

$$\ddot{\mathbf{x}}_c = \ddot{\mathbf{x}}_d + \mathbf{B}_e\dot{\tilde{\mathbf{x}}} + \mathbf{K}_e\tilde{\mathbf{x}} \tag{6}$$

where $\mathbf{B}_e, \mathbf{K}_e$ describe the desired task space impedance, $\tilde{\mathbf{x}} = \mathbf{x}_d - \mathbf{x}$ describes the task error, and $\ddot{\mathbf{x}}_c$ is the task-space command acceleration. To project this control law into the joint space, so it can be substituted into (5), we need to devise a suitable $m \times n$ task Jacobian $\mathbf{J}(\mathbf{q})$ such that

$$\dot{\mathbf{x}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \tag{7}$$

where $\mathbf{x}$ is the $m \times 1$ task state trajectory vector (in this transport scenario, $m = 3$), and n is the length of the joint state vector $\mathbf{q}$.

We note that the standard Jacobian transpose mapping between torques and forces is incomplete for redundant manipulators in motion. Because of the system redundancy, at any given configuration there is an essentially infinite set of joint torque vectors that could theoretically be applied without affecting the resulting end-effector forces. However, for a real robot with non-zero inertia, only one generalised Jacobian inverse is consistent with the system dynamics [17]:

$$\mathbf{J}^{\#}(\mathbf{q}) = \mathbf{M}^{-1}\mathbf{J}^{T}(\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^{T})^{-1} \tag{8}$$

By calculating a corresponding task inertia matrix Λ_x [16], the required task space acceleration can now be projected to a joint torque command:

$$\boldsymbol{\tau}_{task} = \mathbf{J}^{T}\left(\Lambda_x\left(\ddot{\mathbf{x}}_d + \mathbf{B}_e\dot{\tilde{\mathbf{x}}} + \mathbf{K}_e\tilde{\mathbf{x}}\right) + \mathbf{J}^{\#T}\boldsymbol{\tau}_e\right) \tag{9}$$

where $\boldsymbol{\tau}_e$ is an estimate of the external torque.

2) Disturbance observer: We use the task space error and (controllable, known) joint impedance to calculate an estimate of the external torque:

$$\boldsymbol{\tau}_e = \mathbf{K}_\nu\mathbf{J}^{\#}\tilde{\mathbf{x}} \tag{10}$$

3) Null space control: For controlling the null dimensions, we choose a low-stiffness impedance control law:

$$\dot{\boldsymbol{\nu}}_c = -\Lambda_\nu^{-1}((\boldsymbol{\mu}_\nu + \mathbf{B}_\nu)\boldsymbol{\nu} - \mathbf{Z}^{T}\mathbf{K}_\nu\tilde{\mathbf{q}}) \tag{11}$$

The null space stiffness $\mathbf{K}_\nu$ is equivalent to the internal stiffness on the joint impedance drivers, and $\mathbf{B}_\nu$ describes a suitable null-space damping matrix [18]. Λ_ν, $\boldsymbol{\mu}_\nu$ are the null-space inertia matrix and Coriolis matrix [16]. The null space state velocity vector $\boldsymbol{\nu}$ must be derived from the task space to joint space decomposition. For a task-space command with m control dimensions, the impedance behaviour projected into the null space gives us n equations with $r = n - m$ dimensions, meaning that the null space equations are not all independent. To overcome this, we introduce an $(n \times r)$ matrix $\mathbf{Z}$ [18] such that $\mathbf{J}\mathbf{Z} = \mathbf{0}$. We choose $\mathbf{Z} = \begin{bmatrix}\mathbf{J}_r^{T}\mathbf{J}_m^{-T} & \mathbf{I}\end{bmatrix}^{T}$ where $\mathbf{J}_m, \mathbf{J}_r$ are the submatrices of a Jacobian partition $\mathbf{J} = [\mathbf{J}_m \quad \mathbf{J}_r]$ such that $\mathbf{J}_m$ is full rank and invertible. We can now establish a set of null space variables $\boldsymbol{\nu}$ and an extended Jacobian matrix $\mathbf{J}_E(\mathbf{q})$ such that

$$\begin{pmatrix}\dot{\mathbf{x}}\\ \boldsymbol{\nu}\end{pmatrix} = \mathbf{J}_E(\mathbf{q})\dot{\mathbf{q}} = \begin{pmatrix}\mathbf{J}(\mathbf{q})\\ \mathbf{Z}^{\#}(\mathbf{q})\end{pmatrix}\dot{\mathbf{q}} \tag{12}$$

The joint control torque corresponding to the desired null space impedance behaviour can now be calculated:

$$\boldsymbol{\tau}_{null} = -\mathbf{Z}^{\#T}\left(\Lambda_\nu\dot{\mathbf{Z}}^{\#}\dot{\mathbf{q}} + (\boldsymbol{\mu}_\nu + \mathbf{B}_\nu)\boldsymbol{\nu} - \mathbf{Z}^{T}\mathbf{K}_\nu\tilde{\mathbf{q}}\right) \tag{13}$$

C. State observer and adaptive stiffness control

For a system with a highly accurate internal dynamic model handling a known load, and with only constant external disturbance forces, the above control law would be stable and sufficient. However, we find that if the robot's inertial estimates are inaccurate (e.g., when the robot is asked to manipulate an unknown load), or the disturbance is dynamic (e.g., the number of agents in the system varies during the task), this control law may have suboptimal task-space accuracy and convergence speed (see Results, Figure 5). Taking advantage of the variable stiffness tuning offered by modern force-sensitive manipulators, we can implement an adaptive stiffness control that adjusts the compliance in the null space according to the task space performance.

Consider equation (13). We see that joint space impedance is chiefly driven by the joint space error term $\tilde{\mathbf{q}}$ (modulated by the desired joint stiffness $\mathbf{K}_\nu$). A task-state observer can be used to estimate and update the goal joint state according to the instantaneous system dynamics and goal task state:

$$\mathbf{q}_d = \mathbf{q} + \alpha\mathbf{J}^{\#}\tilde{\mathbf{x}} \tag{14}$$

($\mathbf{q}$ is the current joint state, α is some small scaling constant proportional to the control loop time step, β is a scalar gain term). We now replace the constant stiffness matrix $\mathbf{K}_\nu$ with an adaptive diagonal joint stiffness matrix $\mathbf{K}_{var}(t)$:

$$\mathbf{K}_{var}^{i,i}(t) = k_\nu + \left(\frac{\beta\tilde{q}_i(t)}{1 + \dot{\tilde{q}}_i(t)}\right)^2 \tag{15}$$

where $\tilde{q}_i$ represents the projected state error for joint i and k_ν is a minimum equilibrium stiffness (corresponding to the null state stiffness chosen earlier).

When the task error becomes large, the joints "stiffen" until the robot returns to a controllable state, in the same way a human performing a carrying task might stiffen their joints in response to slippage or disturbance, until reaching a more controllable configuration.

D. Leader-follower collective transport

Once multiple agents following the above control schema have a secure grasp on a shared load, we can examine how this system can be used for collaborative transport.

The leader applies a force on the shared load in the direction of a goal position in the plane. Through the body of the load, this force is imparted to the manipulators at the grip points. Appropriate force-sensing at the grip points (e.g., thin-film resistive force sensors) could disambiguate the direction and type of the resultant localised forces. Alternatively, by permitting the grippers to move freely in the lateral plane, a robot can use the relative distance traveled by the robot end effector as a proxy to estimate forces ($\mathbf{F}_{est} = k_{env} * \Delta\mathbf{x}$). The latter method significantly reduces the processing required, as extracting reliable and relevant force data from contact sensors is not always trivial. The inertia of the load itself also acts as a low-pass filter on the transmitted signal.

E. Mobile base outer control loop

The mobile base control algorithm seeks to maintain a constant relative position between end-effector and base centroid, by moving the base in accordance with force or position error signals relayed by the manipulator. In simulation, we assume we have full control over the mobile base trajectory and implement this control as a simple position-based PID loop:

$$\mathbf{F}_m = K_m\mathbf{e} + B_m\dot{\mathbf{e}} \tag{16}$$

where $\mathbf{F}_m$ is the force driving the mobile base, K_m, B_m are the chosen PID gains, $\mathbf{e} = (x(t), y(t)) - (x_0, y_0)$, and (x_0, y_0) represents a starting or neutral position of the end-effector in the mobile base frame. Choosing high damping and low displacement gains minimises overshoot which could lead to oscillatory behaviour at the end-effector.

V. Experimental Platform

The experimental platform used for validation and testing was the 7DoF active compliant manipulator Franka Emika Panda [19]. We created a simulation of this platform in the Unity engine, including accurate dynamic and kinematic models and active impedance drivers on all joints [20]. To test the transport algorithm, multiple instances of the Panda manipulator were mounted on planar mobile bases.

The individual agent control was validated on a single robot with (a) a holding task (Section VI-A), where the goal was to maintain a task space position in the presence of external disturbances, and (b) a trajectory-following task (Section VI-B), where the robot attempted to move along a predefined task-space trajectory while supporting an unknown mass, near the payload limit of the manipulator.

Finally, to demonstrate how this decoupling of manipulator compliance can be used to effect collective transport, we implemented a transportation task using four robots and a load represented by a heavy (20 kg) dining table with struts suitable for secure grasping (Fig. 3, Section VI-C). The simulation is initialized with the table already in the robots' grasp. Each robot seeks to hold the task-space variables (z, θ, ϕ) steady, and transmit any end-effector motion in the (x, y) plane to the mobile platform controller. The leader does not apply any stabilising forces, but adds a lateral force, bounded at 40N, in the direction of a target destination, in accordance with the proportional-distance controller described earlier. The robots stabilising the load have no knowledge of each other's poses or the waypoint position, nor any direct communication with the leader.

All experiments were undertaken using the following control parameter values (where appropriate): $\mathbf{B}_e = \text{diag}\,[1, 1, 1]$, $\mathbf{K}_e = \text{diag}\,[80, 80, 80]$, $\mathbf{B}_\nu = \text{diag}\,[0.4, 0.4, 0.4, 0.4]$, $\mathbf{K}_\nu = \text{diag}\,[15, 5, 5, 5, 5, 5, 5]$, $B_m = 20$, $K_m = 2000$.

VI. Results

A. Validation: Selective control of the end-effector space

Disturbance forces during transportation may include dynamic or abrupt external torques. Fig. 4 shows the response of the manipulator to (top) smoothly varying and (bottom) stochastic and abrupt external disturbances at the end-effector. In both cases, the robot's task space consists of the (z, θ, ϕ) end-effector dimensions. The disturbance forces induce motion in the other (null-space) dimensions, but the robot maintains task space accuracy even with the application of sudden and unpredictable forces.

B. Adaptive stiffness

Fig. 5 shows the utility of dynamic stiffness, plotting the accuracy of a trajectory-following task when the robot is supporting an unknown load of significant mass. Without knowledge of the inertial properties of the payload or any compensatory adaptive behaviour, the accuracy of the task-following is impaired. Adding a dynamic stiffness term not only improves the task space accuracy, it also reduces the overall torque exerted by the joint actuators. Although the stiffness term is quadratic (so that higher internal joint forces might be expected), the corresponding improvement in task space error means that the total joint forces resulting from both task- and null-space control torques are significantly lower with the adaptive term in place, and also converge more speedily to a stable steady-state force application.

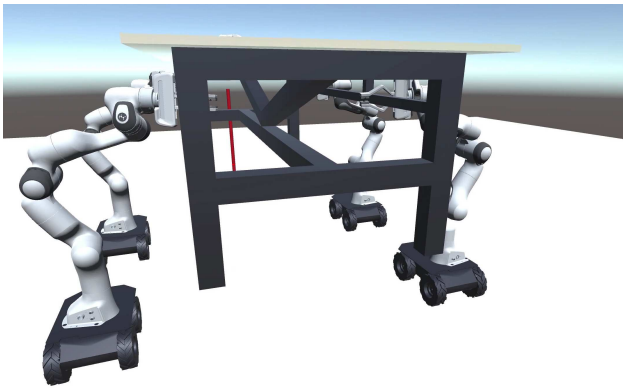

Fig. 3. Four simulated robots hold and transport a 20kg table, not modified to facilitate manipulation by robots. See accompanying video.

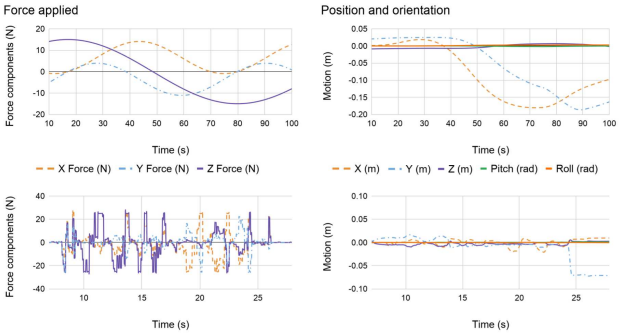

Fig. 4. Non-constant external forces applied to the Panda end-effector. Left: external Cartesian forces applied; right: the corresponding end-effector pose (x, y, z, θ, ϕ).

C. Collective transport

To test the performance of the described algorithm in a collaborative transport task, we examine the motion of a heavy payload (Fig. 3) being transported by four autonomous robotic agents, following guidance forces imparted by a (disembodied) leader. The leader applies a force at one end of the table in the direction of a goal whose location is not known to the follower agents. The followers attempt to stabilize the load in the local task space (z, θ, ϕ), while using sensed motion of the end-effector in the (x, y) plane to control the motion of the mobile base.

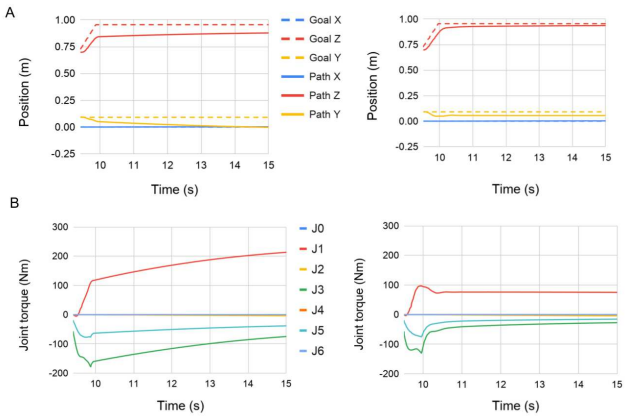

Fig. 5. Joint forces and task space performance without (left) and with (right) dynamic joint stiffness tuning. (A) Trajectory-following performance. Dotted lines represent the desired trajectory, solid lines represent the robot end-effector position. (B) Joint forces. Dynamic stiffness control ($\beta = 100$) allows for smaller position error and smaller overall joint forces, with faster convergence to steady-state torque application.

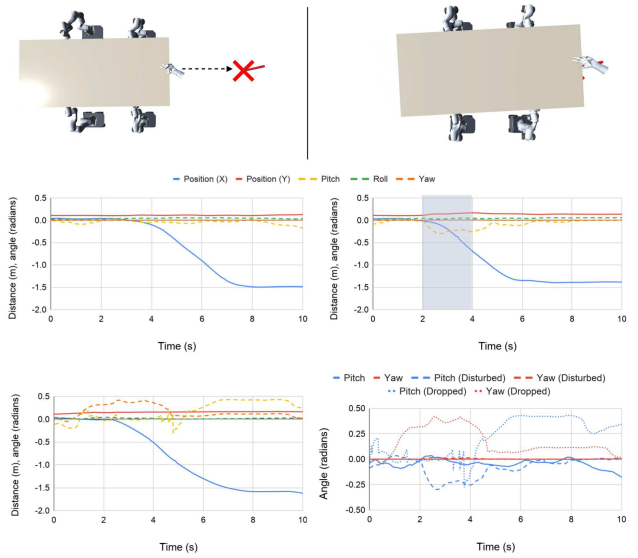

Fig. 6. The motion of the centre of mass of a large payload during a collective transport task with four independent agents and an external guiding force (represented by a hand at the force application point), with target location shown by a red X. Top: Start state and end state of the given transport task. Middle left: Motion of the table with no external disturbance on the system. Middle right: Motion of the table when one agent has a large (50N) force exerted on the end-effector for 2s (shaded area). Bottom left: Motion of the table when one agent fails to grasp the table. Bottom right: A closer examination of the out-of-plane rotations experienced by the table under different transport conditions. Solid lines: no disturbance; dashed line: external force for 2s; dotted line: missing agent. See also video attachment.

Fig. 6 and the video show the behaviour of the centre of mass of the payload table under different transport conditions. In the first, no external disturbances are introduced, and we see the payload move smoothly to the goal location. In the second, an external force of 50N is applied for a short time to the end-effector of one of the follower agents, in the horizontal plane and perpendicular to the direction of motion, to simulate an impact or collision. Some minor out-of-plane rotation is experienced at the centre of the table during the force application, but it quickly returns to the initial pose after the force is removed. Part 2 of the video attachment also shows the performance of the system with transient planar perturbation of a single robot in both perpendicular and parallel directions, and with an in-plane torque. In the third demonstration, an agent fails its initial grasp on the table. This disturbance entails brief spurious forces on the load during the first $\sim$1 second and then, after the robot's grasp slips away entirely, a loss of 25% of the carrying capacity of the system, resulting in a very uneven load distribution for the remaining agents. While the loss of an agent has a noticeable effect on both in-plane and out-of-plane rotation, the payload pose remains within controllable bounds and the transportation task is still accomplished quickly and smoothly.

VII. Discussion and Future Work

We present a framework for collaborative transport, in which an arbitrary number of independent robots help an informed agent move a shared load in a desired way, while requiring minimal system knowledge on the part of individual agents. Force sensitivity (whether directly sensed, or inferred from movement of the load) and adaptive stiffness allow implicit communication and fast, intuitive handling of disturbances in a process analogous to that seen in physical human collaboration.

The ability to decouple compliance in the end-effector operational space means that as long as the broad parameters of the task are known in advance (i.e., in which task-space dimensions the agent can expect to receive a guiding force signal), disturbances and inertial wrenches from the payload can be successfully rejected, while guiding signals can be sensed at the end-effector and transmitted to an outer navigational control loop. We also show that in the case of a heavy payload, one which causes significant mismatch between the robot's internal model and the real system behaviour, an adaptive stiffness control at the joint level will enable the robot to rectify its performance without needing to explicitly estimate the inertial payload parameters. Together, these control elements result in a robust collaborative transport algorithm which can cope with significant system disturbances.

We note that for this work, each task domain is explicitly declared in the individual robots' control algorithm. However the inherent force sensitivity of the manipulators opens the way for a task encoding framework which could allow the leader to signal a switch between constrained dimensions via force application, particularly if tactile force sensing is added to the follower's grippers (eg. a vibrational signal transmitted through the collective load). Immediate future research includes developing such a framework, including a real-world system demonstration on a realistic problem scenario, and providing the leader with a control algorithm which incorporates force sensing at the payload as well as position feedback, allowing adaptive trajectory generation for energy-optimal navigation of complex environments.

REFERENCES

[1] S. Berman, Q. Lindsey, M. S. Sakar, V. Kumar, and S. C. Pratt, "Experimental study and modeling of group retrieval in ants as an approach to collective transport in swarm robotic systems," *Proceedings of the IEEE*, vol. 99, no. 9, pp. 1470–1481, 2011.

[2] G. Kumar, A. Buffin, T. Pavlic, S. Pratt, and S. Berman, "A stochastic hybrid system model of collective transport in the desert ant aphaenogaster cockerelli," in *HSCC 2013—Proceedings of the 16th International Conference on Hybrid Systems*, Dec. 2012, pp. 119–124.

[3] M. Rubenstein, A. Cabrera, J. Werfel, G. Habibi, J. McLurkin, and R. Nagpal, "Collective transport of complex objects by simple robots: theory and experiments," in *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-Agent Systems*, 2013, pp. 47–54.

[4] G. Habibi, Z. Kingston, W. Xie, M. Jellins, and J. McLurkin, "Distributed centroid estimation and motion controllers for collective transport by multi-robot systems," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 1282–1288.

[5] R. Brown and J. Jennings, "A pusher/steerer model for strongly cooperative mobile robot manipulation," in *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, vol. 3. Pittsburgh, PA, USA: IEEE Comput. Soc. Press, 1995, pp. 562–568. [Online]. Available: http://ieeexplore.ieee.org/document/525941/

[6] Z.-D. Wang, Y. Takano, Y. Hirata, and K. Kosuge, "A pushing leader based decentralized control method for cooperative object transportation," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 1. Sendai, Japan: IEEE, 2004, pp. 1035–1040. [Online]. Available: http://ieeexplore.ieee.org/document/1389489/

[7] A. Petitti, A. Franchi, D. Di Paola, and A. Rizzo, "Decentralized motion control for cooperative manipulation with a team of networked mobile manipulators," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. Stockholm, Sweden: IEEE, May 2016, pp. 441–446. [Online]. Available: http://ieeexplore.ieee.org/document/7487164/

[8] G. Montemayor and J. Wen, "Decentralized Collaborative Load Transport by Multiple Robots," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. Barcelona, Spain: IEEE, 2005, pp. 372–377. [Online]. Available: http://ieeexplore.ieee.org/document/1570147/

[9] H. Bai and J. T. Wen, "Cooperative Load Transport: A Formation-Control Perspective," *IEEE Transactions on Robotics*, vol. 26, no. 4, pp. 742–750, Aug. 2010. [Online]. Available: http://ieeexplore.ieee.org/document/5504175/

[10] A. Franchi, A. Petitti, and A. Rizzo, "Decentralized parameter estimation and observation for cooperative mobile manipulation of an unknown load using noisy measurements," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. Seattle, WA, USA: IEEE, May 2015, pp. 5517–5522. [Online]. Available: http://ieeexplore.ieee.org/document/7139970/

[11] A. Marino, G. Muscio, and F. Pierri, "Distributed cooperative object parameter estimation and manipulation without explicit communication," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. Singapore, Singapore: IEEE, May 2017, pp. 2110–21 116. [Online]. Available: http://ieeexplore.ieee.org/document/7989243/

[12] Z. Wang and M. Schwager, "Kinematic multi-robot manipulation with no communication using force feedback," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. Stockholm, Sweden: IEEE, May 2016, pp. 427–432. [Online]. Available: http://ieeexplore.ieee.org/document/7487163/

[13] H. Farivarnejad, S. Wilson, and S. Berman, "Decentralized sliding mode control for autonomous collective transport by multi-robot systems," in *2016 IEEE 55th Conference on Decision and Control (CDC)*. Las Vegas, NV, USA: IEEE, Dec. 2016, pp. 1826–1833. [Online]. Available: http://ieeexplore.ieee.org/document/7798530/

[14] B. Hichri, J. Fauroux, L. Adouane, I. Doroftei, and Y. Mezouar, "Design of cooperative mobile robots for co-manipulation and transportation tasks," *Robotics and computer-integrated manufacturing*, vol. 57, pp. 412–421, 2019.

[15] S. De Serres and T. Milner, "Wrist muscle activation patterns and stiffness associated with stable and unstable mechanical loads," *Experimental brain research*, vol. 86, no. 2, pp. 451–458, 1991.

[16] H. Sadeghian, L. Villani, M. Keshmiri, and B. Siciliano, "Task-space control of robot manipulators with null-space compliance," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 493–506, 2013.

[17] O. Khatib, L. Sentis, J. Park, and J. Warren, "Whole-body dynamic behavior and control of human-like robots," *International Journal of Humanoid Robotics*, vol. 1, no. 01, pp. 29–43, 2004.

[18] C. Ott, *Cartesian impedance control of redundant and flexible-joint robots*. Springer, 2008.

[19] C. Gaz, M. Cognetti, A. Oliva, P. R. Giordano, and A. De Luca, "Dynamic identification of the franka emika panda robot with retrieval of feasible parameters using penalty-based optimization," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4147–4154, 2019.

[20] N. Carey, "Unity implementation of the franka panda platform," 2020. [Online]. Available: https://github.com/niccarey/FrankaPanda_Unity