A decentralized cooperative transportation scheme for humanoid robots

Greta Gasbarrone, Nicola Scianca, Leonardo Lanari, and Giuseppe Oriolo

***Abstract*— We propose an MPC-based decentralized scheme for cooperative transportation between two agents. One agent, the *leader*, can be either a human or a robot with knowledge of the task. The other, the *follower*, has no knowledge of the task, and must autonomously decide how to move based on the perceived interaction forces. The robots interact with the object in a compliant way thanks to a hand admittance controller, and the follower continuously adapts its footstep plan in order to accommodate the hand displacement. The combination of these two effects allows the follower to smoothly react to the motion of the leader: it can move omnidirectionally and rotate, as well as accommodate lifting and lowering of the transported object, all while performing obstacle avoidance during footstep placement. We report dynamic simulations on two HRP-4 robots in a number of different scenarios, both when carrying a table and an object with handles.**

I. Introduction

Facilitating the introduction of robots in fields that involve manual labor can bring about many advantages, among which is the reduction the amount of heavy work that must be done by humans. This reduction can be achieved by either completely entrusting the heavier tasks to robots, or by using robots to collaborate with humans.

Collaborative approaches between humanoid robots can make use of a *centralized* controller, which, being able to act on both systems concurrently, is well suited to determine the individual behaviors necessary to achieve the common goal. However, for retaining high flexibility in the range of possible applications, it is better to use a *decentralized* approach which can be applied both to human-robot or to robot-robot scenarios.

In a decentralized approach, two agents are usually identified by the roles that they take on during the transportation [1]. The *leader* has knowledge of the task to be executed, and can be embodied either by a human or, as in our case, by a robot with a predefined plan. The *follower* is a robot that is unaware of the task and must determine how to move based on the interaction forces that it perceives.

Earlier works have explored transportation by pushing, e.g., in [2], where six-legged robots are used to push a box, or in [3], in which mobile robots are used. Other works, such as [4], make use grippers to hold on to the transported object. [5] proposed an approach based on Model Predictive Control (MPC) for human-robot transportation using a mobile-base manipulator. A comprehensive review that includes several classic and more recent works can be found in [6].

The authors are with the Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma, via Ariosto 25, 00185 Roma, Italy. E-mail: {lastname}@diag.uniroma1.it.

Nicola Scianca has been fully supported by PNRR MUR project PE0000013-FAIR.

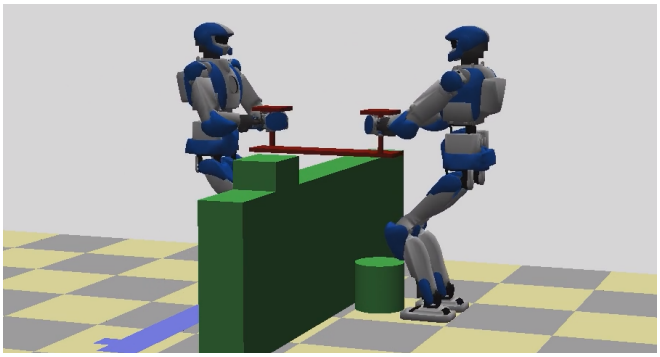

Fig. 1. Cooperative transportation of an object over a tall obstacle. Only the leader (left) has information about the task, while the follower (right) reacts to the motion of the transported object on the basis of the perceived interaction forces.

Focusing on humanoid robots restricts the scope to a much more contained number of works. In [7], two humanoids perform transportation of a table along a straight line path using a centralized approach. [8] tackles human-humanoid transportation by focusing more on the problem of human intention detection rather than on the robot control aspect, which is approached by separately controlling the upper and lower body. [9] has proposed a walking pattern generator in which role-specific objective functions determine the robot behaviour in human-robot collaborative tasks along a straight line. In [10], the authors use a combination of hand admittance and predictive control, with motions limited to straight-line walking in an obstacle free environment. In [11], some curve-line paths are achieved by combining different behaviors thanks to a finite state machine, which however, due to its intrinsically discrete nature, cannot blend these behaviors together.

In this paper, we propose a decentralized scheme in which controllers for both the leader and the follower are based on the Intrinsically Stable Model Predictive Control (IS-MPC) [12]. IS-MPC simultaneously determines the trajectory of the Center of Mass (CoM) and of the Zero Moment Point (ZMP). The ZMP is the point of application of the ground reaction force, which guarantees contact stability as long as it is kept within the support polygon (i.e., the convex hull of contact surfaces). To guarantee boundedness of the generated trajectories, IS-MPC features a *stability constraint*, which we showed to be useful also for performing disturbance rejection [13], an aspect that is particularly relevant in this setting where the weight of the transported object and the interaction forces cannot be ignored.

Our approach uses a *hand admittance controller*, which allows both robots to hold on to the object in a compliant

way. The follower robot, having no knowledge of the task, determines its footstep placement so as to recuperate the hand displacement generated by the admittance. The combination of admittance and footstep replanning is what make the follower move in reaction to a motion of the leader.

The range of possible motions that can be generated in this way is very rich, including omnidirectional walking (i.e., frontal, lateral, and diagonal), as well as changes in footstep orientation. The follower is placing footsteps using a constrained optimization problem, which allows us to perform obstacle avoidance by forbidding stepping into particular regions.

Compared with the state of the art, our approach offers several improvements:

- by using a three-dimensional admittance-based approach, we are able to lift and lower the transported object, e.g., to overcome an obstacle or to better adapt to changes in stature of a human transporter, for the case of human-robot collaboration;
- we design a continuous (rather than rule-based) strategy adapting the orientation of the footsteps, overcoming a common limitation in the literature where the transportation usually takes place along a straight line;
- we plan footsteps in a dedicated module separate from the gait generation, which allows us to perform more sophisticated planning (e.g., involving nonlinear and nonconvex constraints) without burdening the MPC, which is always linear and can comfortably run at high frequency.

The paper is organized as follows. Section II gives an overview of the control architectures of the leader and the follower. In Sect. III, we introduce the dynamic model that will be used by the predictive controller. The hand admittance controller is described in Sect. IV, while the footstep planner for the follower is presented in Sect. V. The MPC used for gait generation is illustrated in Sect. VI. Simulation results for different transportation tasks are shown in Sect. VII, while some concluding remarks are offered in Sect. VIII.

II. Control Architecture

The goal of the proposed scheme is to have two agents collaborate in transporting an object using a decentralized approach, so that it can be applied both to human-robot and robot-robot scenarios. In our case, the leader is a robot who is directly instructed about the transportation task. In particular, the description of the task is given in the form of an assigned trajectory of the object, which the leader uses to plan a footstep sequence. The follower robot has no information about the task, but will experience interaction forces through the object as soon as the leader starts moving. We intend to give the follower the ability to react to these interaction forces in a way that (i) safeguards balance, so that the robot does not fall and (ii) allows to maintain the grasp of the object at all times, so that it can effectively collaborate in the transportation.

Figure 2 shows a block scheme of the proposed control architecture for both the leader and the follower. We assume that both robots start from a bimanual grasping configuration, that is maintained throughout the entire motion. For each robot, we measure or estimate the force applied at each hand. In an experimental setting, this would be done either using a force sensor at the wrist [9], [14] or by reconstructing the interaction force, e.g., with a residual-based technique [15].

In both the leader and the follower, the perceived interaction forces are fed to an *admittance controller*, which regulates the position of each hand around a reference value according to a mass-spring-damper model. Such positions are then sent to the whole-body robot controller.

The *footstep planner* module is substantially different for the two agents: in particular, the leader decides its plan solely based on the transportation task, whereas the follower generate its footsteps in reaction to the average hand displacement with respect to their reference positions. To this end, we propose an optimization-based planner which can also handle non-convex constraints for obstacle avoidance.

The combination of the admittance control and the footstep planner allows the follower robot to react to the interaction forces caused by the leader by using two concurrent effects: initially, it reacts by moving the hands, but since this also triggers a replanning of the footsteps, the entire body will follow. We find that having the robot body react indirectly is preferable to directly moving based on the perceived force. In fact, this strategy inherently gives the hands a priority over the body, smoothing out the effects of variations in the interaction forces and filtering the inevitable noise affecting their measurements.

Finally, both the leader and the follower use an MPC controller for gait generation based on their footstep plan. In particular, we adopt the *IS-MPC* algorithm [12], which determines the CoM trajectory in such a way that dynamic balance and internal stability are guaranteed throughout the motion. More precisely, we employ a robust variant of IS-MPC [13], which relies on a disturbance observer to estimate perturbations on the CoM, and injects this information in a robust stability constraint so as to perform disturbance compensation.

The hand trajectories resulting from the admittance controller and the CoM trajectory produced by IS-MPC are finally sent to the whole-body controller, which generates joint acceleration commands for driving the robot.

III. Dynamic Model

In this section, we derive a centroidal model for a humanoid of mass m transporting an object with two hands.

One starts by imposing the balance of moments with respect to the point of application of the ground reaction force, which removes the contribution of the latter. The remaining forces are (1) gravity $m\boldsymbol{g} = (0, 0, -mg)$, acting on the CoM, and (2) external forces $\boldsymbol{f}_i = (f_i^x, f_i^y, f_i^z)$, $i = l, r$, respectively applied on the left/right hand, whose positions in the world frame are denoted as $\boldsymbol{p}_i = (x_i, y_i, z_i)$. We obtain

$$\dot{\boldsymbol{L}} + (\boldsymbol{p}_c - \boldsymbol{p}_z) \times m\ddot{\boldsymbol{p}}_c = (\boldsymbol{p}_c - \boldsymbol{p}_z) \times m\boldsymbol{g} + \sum_{i=l,r} (\boldsymbol{p}_i - \boldsymbol{p}_z) \times \boldsymbol{f}_i$$

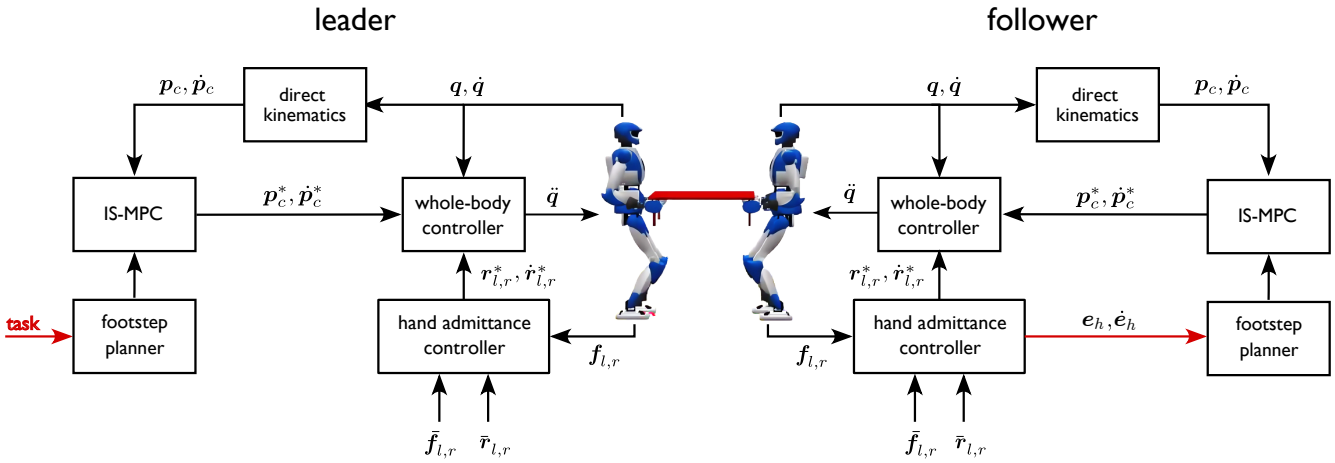

Fig. 2. Control architectures for the leader and the follower, with role-specific signals shown in red. The two robots exchange forces through the transported object but no other information. As a consequence, the scheme will still work if the leader is replaced by a human.

where $\boldsymbol{p}_c = (x_c, y_c, z_c)$ is the CoM position, $\boldsymbol{p}_z = (z_x, z_y, 0)$ is the ZMP position, and $\boldsymbol{L}$ is the centroidal angular momentum. We can rewrite the above expression as

$$\dot{\boldsymbol{L}} + (\boldsymbol{p}_c - \boldsymbol{p}_z) \times (m\ddot{\boldsymbol{p}}_c - m\boldsymbol{g} - \boldsymbol{f}) = \sum_{i=l,r} (\boldsymbol{p}_i - \boldsymbol{p}_c) \times \boldsymbol{f}_i,$$

with $\boldsymbol{f} = \sum_{i=l,r} \boldsymbol{f}_i$. From this, denoting by $\boldsymbol{v}^{xy}$ the subvector consisting of the first two components of a 3D vector $\boldsymbol{v}$, one obtains

$$\ddot{\boldsymbol{p}}_c^{xy} = \frac{m(g+\ddot{z}_c) - f^z}{mz_c}(\boldsymbol{p}_c^{xy} - \boldsymbol{p}_z^{xy}) + \frac{\boldsymbol{S}(\boldsymbol{n}^{xy} - \dot{\boldsymbol{L}}^{xy}) + z_c \boldsymbol{f}^{xy}}{mz_c} \tag{1}$$

where $\boldsymbol{S}$ is the rotation matrix by $-\pi/2$, $f^z = \sum_{i=l,r} f_i^z$, and $\boldsymbol{n} = \sum_{i=l,r} (\boldsymbol{p}_i - \boldsymbol{p}_c) \times \boldsymbol{f}_i$. If we assume a constant CoM height $\bar{z}_c$, and collect in a disturbance term $\boldsymbol{w}^{xy}$ the second term in the right-hand side term of (1), we get the *perturbed Linear Inverted Pendulum* (LIP) model

$$\ddot{\boldsymbol{p}}_c^{xy} = \eta^2 (\boldsymbol{p}_c^{xy} - \boldsymbol{p}_z^{xy}) + \boldsymbol{w}^{xy}, \tag{2}$$

where the natural pendulum frequency

$$\eta = \frac{\sqrt{mg - f^z}}{m\bar{z}_c}$$

accounts for the vertical component f_z of the total force $\boldsymbol{f}$ applied on the hands. Note that the disturbance depends on (1) the horizontal components $\boldsymbol{f}^{xy}$ of $\boldsymbol{f}$ (2) the sum $\boldsymbol{n}$ of the centroidal moments of the two external forces, and (3) the angular momentum variation, which is related to the whole-body dynamics.

In order to make the linear model (2) time-invariant for control design, we will use in η a nominal value $\bar{f}^z$ of f^z, equal to half the weight of the transported object. The effect of perturbations with respect to $\bar{f}^z$ will be lumped into the disturbance term.

IV. Hand Admittance Controller

We now begin the description of the different components of the proposed scheme, starting with the hand position admittance controller.

The hand admittance controller regulates the position of each hand with respect to the robot body in response to the perceived interaction force $\boldsymbol{f}_i$ on the hand, $i = l, r$. For each robot, let us define a local 3D frame $\mathcal{F}$ having the origin at the midpoint[1] between the ground projections of the left and right feet (such projection exists even when the foot is airborne) and orientation $\boldsymbol{R}_{\mathcal{F}}$ equal to the orientation of the current support foot. The 3D positions of the robot hands relative to this frame are denoted as $\boldsymbol{r}_i$, $i = l, r$.

Each hand is assigned a reference grasping position $\bar{\boldsymbol{r}}_i$, which is constant in $\mathcal{F}$ and chosen in such a way to yield a comfortable posture of the arm and a sufficient level of manipulability at the same time. In addition, since it is assumed that both grasps are maintained at all times, the references for the two hands should be chosen so as to satisfy a suitable rigidity constraint in order to avoid the appearance of internal forces, at least at steady state.

The relative hand position $\boldsymbol{r}_i$, $i = l, r$, is determined so as to mimic a virtual mass-spring-damper system loaded by the external force $\boldsymbol{f}_i$, with rest position $\bar{\boldsymbol{r}}_i$ when the external force is $\bar{\boldsymbol{f}}_i$:

$$\boldsymbol{M}\ddot{\boldsymbol{r}}_i + \boldsymbol{C}\dot{\boldsymbol{r}}_i + \boldsymbol{K}(\boldsymbol{r}_i - \bar{\boldsymbol{r}}_i) = \boldsymbol{R}_{\mathcal{F}}^T(\boldsymbol{f}_i - \bar{\boldsymbol{f}}_i),$$

where the diagonal matrices $\boldsymbol{M}$, $\boldsymbol{C}$ and $\boldsymbol{K}$ contain the mass, damping and stiffness along the three axes. While we assume that these matrices are the same for the left and the right hand, they can be different for the leader and the follower, in view of their different role in transporting the object. Note that the reference force on each hand at rest is $\bar{\boldsymbol{f}}_i = (0, 0, \bar{f}^z/2)$, $i = l, r$, corresponding to a quarter of the weight of the object.

V. Footstep Planner

Footstep generation is inherently different for the two robots. For the leader, the footstep plan is chosen so as to

[1]Another possibility would be to place the local frame at the CoM or at a fixed point on the robot, e.g., the torso. A drawback of these choices is that $\mathcal{F}$ would inherit the lateral sway motion associated with the gait.

realize the given transportation task; in our implementation, the task was specified through a desired velocity for the object and footsteps were generated using a planner similar to the one described in [12]. The follower, however, has no direct information about the task: its footstep plan generation is then essentially reactive and aimed at maintaining the grasp on the object as the latter is moved by the leader. In addition, both the leader and the follower should also avoid collisions with workspace obstacles. In the following, we focus on the footstep planner used by the follower.

Footstep planning is performed in real time and incrementally: that is, at each instant of time the next F footsteps are generated by choosing their position $\boldsymbol{p}_f^j = (x_f^j, y_f^j)$ and orientation θ_f^j, $j = 1, \ldots, F$, while the step duration T_{step} is assumed to be fixed throughout the plan. In particular, footstep placement is found by solving a constrained optimization problem which makes use of two nominal displacements $\Delta\boldsymbol{p}$ and $\Delta\theta$, whose determination is discussed in detail below.

The nominal Cartesian displacement $\Delta\boldsymbol{p} = (\Delta x, \Delta y)$ between successive footsteps in the plan should make the body of the robot move in accordance with its hands. To this end, define the *average hand error*

$$\boldsymbol{e}_h = \begin{pmatrix} e_h^x \\ e_h^y \end{pmatrix} = \frac{\bar{\boldsymbol{r}}_l^{xy} - \boldsymbol{r}_l^{xy} + \bar{\boldsymbol{r}}_r^{xy} - \boldsymbol{r}_r^{xy}}{2},$$

and let

$$\begin{aligned} \Delta x &= k_{p,x} e_h^x + k_{d,x} \dot{e}_h^x + k_{i,x} \int_0^t e_h^x \, dt \\ \Delta y &= k_{p,y} e_h^y + k_{d,y} \dot{e}_h^y + k_{i,y} \int_0^t e_h^y \, dt, \end{aligned}$$

with all gains positive. The structure of these formulas is clearly reminiscent of a PID controller, with the integral term needed to ensure that the follower robot produces a persistent gait at steady state, i.e., when $\boldsymbol{e}_h$ has converged to zero (consider that the horizontal interaction forces can be expected to oscillate around zero when the object is being transported at a constant velocity). The derivative term is obviously introduced to dampen the response.

Coming to the nominal angular displacement $\Delta\theta$, its objective is to allow the follower to curve when the leader does. Define then the *transportation axis* as the line through the origin of $\mathcal{F}$ which is normal to the segment joining $\boldsymbol{r}_l^{x,y}$ and $\boldsymbol{r}_r^{x,y}$ (the ground projections of the two hands). $\Delta\theta$ is then defined as the orientation of the transportation axis measured with respect to the x axis of $\mathcal{F}$, i.e., the sagittal axis of the robot (see Fig. 3, left). Note that the geometric construction of the transportation axis degenerates when the ground projections of the two hands coincide at a point P, as for example in the object grasp of Fig. 1; in this case, the transportation axis is directly defined as the line joining P with the origin of $\mathcal{F}$ (Fig. 3, right). The degenerate case also allows to execute single-hand (rather than bimanual) transportation tasks.

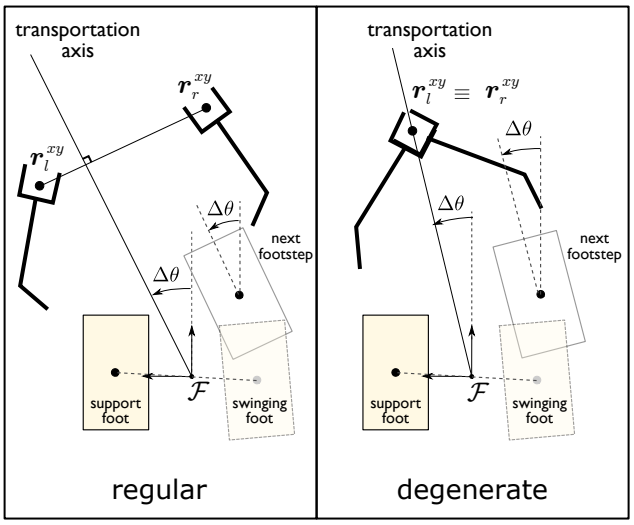

Fig. 3. Geometric construction for determining the transportation axis and the associated angular displacement $\Delta\theta$. Left: regular case. Right: degenerate case (the ground projections of the hands coincide). Besides appearing in the QP which determines the positions of the next F footsteps, $\Delta\theta$ is also used to define the orientation of such footsteps.

Once $\Delta\boldsymbol{p}$ and $\Delta\theta$ have been determined, the position of all the F footsteps in the plan is computed by solving the following quadratic program (QP) with nonlinear constraints:

$$\begin{cases} \min\limits_{\boldsymbol{p}_f^1, \ldots, \boldsymbol{p}_f^F} \sum\limits_{j=1}^{F} \| \boldsymbol{p}_f^j - \boldsymbol{p}_f^{j-1} - \boldsymbol{R}_j (\Delta\boldsymbol{p} \pm (0 \quad \ell)^T) \|^2 \\ \text{subject to:} \\ \bullet \text{ kinematic constraints} \\ \bullet \text{ obstacle avoidance constraints,} \end{cases}$$

where $\boldsymbol{p}_f^0$ is the position of the current support foot (not a decision variable) and we have set

$$\begin{aligned} \boldsymbol{R}_1 &= \boldsymbol{R}_{\mathcal{F}} \\ \boldsymbol{R}_j &= \boldsymbol{R}_{\mathcal{F}} \boldsymbol{R}_{\Delta\theta} \qquad \text{for } j = 2, \ldots, F, \end{aligned}$$

being $\boldsymbol{R}_{\Delta\theta}$ the rotation matrix by $\Delta\theta$.

The cost function of the QP encodes the nominal placement of the planned footsteps: the first footstep $\boldsymbol{p}_f^1$ should be displaced[2] from $\boldsymbol{p}_f^0$ by $\Delta\boldsymbol{p}$, whereas the displacement between the generic successive footstep $\boldsymbol{p}_f^j$ ($j = 2, \ldots, F$) and $\boldsymbol{p}_f^{j-1}$ should be $\Delta\boldsymbol{p}$ rotated by $\Delta\theta$. The actual placement of the footsteps may differ from the nominal placement due to the presence of the constraints, whose structure will be discussed below.

As for the choice of the orientation of the F footsteps, the first footstep of the plan (i.e., the next footstep, corresponding to $j = 1$) is rotated by an angle $\Delta\theta$ with respect to the current support foot, whereas the remaining footsteps ($j = 2, \ldots, F$) have the same orientation of the first.

The role of the kinematic constraints in the QP is to ensure that each footstep is kinematically realizable by the robot.

[2]Note how $\Delta\boldsymbol{p}$ is actually offset in the y direction by $\pm\ell$ to guarantee the appropriate lateral separation between left and right footsteps.

In practice, this is done by defining a rectangular admissible region of dimensions $d_{a,x}$, $d_{a,y}$ aligned with the previous footstep and displaced from its position $\boldsymbol{p}_f^{j-1} = (x_f^{j-1}, y_f^{j-1})$ by a distance $\pm\ell$ in the y direction:

$$\boldsymbol{R}_j^T \begin{pmatrix} x_f^j - x_f^{j-1} \\ y_f^j - y_f^{j-1} \end{pmatrix} \leq \frac{1}{2} \begin{pmatrix} d_{a,x} \\ d_{a,y} \end{pmatrix} \pm \begin{pmatrix} 0 \\ \ell \end{pmatrix} \quad \text{for } j = 1, \ldots, F.$$

The obstacle avoidance constraints in the QP guarantee that the planned footsteps do not lead to a collision between the robot and objects in the environment. In general, these constraints are expressed as

$$\boldsymbol{p}_f^j \in \mathcal{S}, \quad \text{for } j = 1, \ldots, F,$$

where $\mathcal{S} \subseteq \mathbb{R}^2$ denotes the ground region where a footstep can be placed without colliding with obstacles. In the general case, $\mathcal{S}$ will be a non-convex region, leading to a nonlinearly constrained QP. Still, a real-time solution is possible since the number of variables is typically small. In our implementation, we consider a decomposition (possibly by excess) of $\mathbb{R}^2 - \mathcal{S}$ in N_o circular obstacles, and express the obstacle avoidance constraints as

$$\|\boldsymbol{p}_f^j - \boldsymbol{p}_o^l\| \geq \rho^l, \quad \text{for } j = 1, \ldots, F, \quad l = 1, \ldots, N_o,$$

where $\boldsymbol{p}_o^l$ is the center of the l-th obstacle and ρ^l its radius, slightly increased in order to provide a safety margin.

VI. Model Predictive Controller

For both the leader and the follower, the IS-MPC algorithm [12] is used for stable gait generation along the planned footsteps. In this version, IS-MPC does not modify the footstep positions given by the planner. The algorithm operates over sampling intervals of duration δ with a *control horizon* of C samples. It is assumed that $C \leq P$, where $P = F \cdot T_{\text{step}}/\delta$ is the number of sampling intervals included in the footstep plan; in other words, we assume that the footstep plan covers at least the whole control horizon. We denote the generic time instant by t_k and the corresponding value of a generic quantity γ by $\gamma^k = \gamma(t_k)$.

The prediction model for the MPC is a dynamically extended version of the perturbed LIP (2) in which the ZMP velocity $\dot{\boldsymbol{p}}_z^{xy}$ is used as input. This choice allows to include the norm of this velocity as a regularization term in the MPC cost function, ultimately resulting in smoother trajectories.

A sufficient condition for dynamic balance is that the ZMP is at all times inside the *support polygon*, i.e., the convex hull of the contact surfaces. To preserve linearity of the resulting constraint in the presence of foot rotations, we use a slightly conservative condition, which constrains the ZMP to be within a *moving box*, i.e., a region of fixed shape and size $(d_{z,x}, d_{z,y})$, whose center $\boldsymbol{p}_{\text{mb}} = (x_{\text{mb}}, y_{\text{mb}}, 0)$ moves in such a way to always be within the support polygon [16]. In this setting, the ZMP constraint can be written as

$$-\frac{1}{2} \begin{pmatrix} d_{z,x} \\ d_{z,y} \end{pmatrix} \leq \boldsymbol{R}_{\text{mb}}^{k+i} \begin{pmatrix} x_z^{k+i} - x_{\text{mb}}^{k+i} \\ y_z^{k+i} - y_{\text{mb}}^{k+i} \end{pmatrix} \leq \frac{1}{2} \begin{pmatrix} d_{z,x} \\ d_{z,y} \end{pmatrix}$$

$\boldsymbol{R}_{\text{mb}}^{k+i}$ is the rotation matrix associated with the orientation of the moving box at time t_{k+i}.

Dynamic balance by itself, however, does not ensure closed-loop stability, due to the presence of a positive eigenvalue in system dynamics (2). To ensure boundedness between the ZMP and the CoM trajectories, we enforce a *robust stability constraint* along the lines of [13], [12] on the Divergent Component of Motion (DCM) [17]. To derive this constraint, we start from the *stability condition*, which, for the perturbed LIP, is

$$x_u^k = \eta \int_{t_k}^{\infty} e^{\eta(t_k - \tau)} x_z(\tau)\, d\tau - \frac{1}{\eta} \int_{t_k}^{\infty} e^{\eta(t_k - \tau)} w_x(\tau)\, d\tau, \tag{3}$$

along x, and similarly along y. Here, $x_u = x_c + \dot{x}_c/\eta$ is the x-component of the DCM. Condition (3) is *non-causal*, as it depends on future values of the input and disturbance. We write 3 as

$$\eta \int_{t_k}^{t_{k+C}} e^{\eta(t_k - \tau)} x_z(\tau)\, d\tau \;\; = x_u^k - c_x^k + c_{w,x}^k,$$

with

$$\begin{aligned} c_x^k &= \eta \int_{t_{k+C}}^{\infty} e^{\eta(t_k - \tau)} x_z(\tau)\, d\tau \\ c_{w,x}^k &= \frac{1}{\eta} \int_{t_k}^{\infty} e^{\eta(t_k - \tau)} w_x(\tau)\, d\tau \end{aligned}$$

to isolate the sources of non-causality in c_x^k and $c_{w,x}^k$. In particular c_x^k depends on the input outside the control horizon, while $c_{w,x}^k$ is a function on the future disturbance after t_k. To obtain a causal version, we conjecture the future evolution of x_z after the control horizon using the footstep plan. In $c_{w,x}^k$, in place of w_x for $t > t_k$, we use the current estimate $\hat{w}_x^k$ obtained from a disturbance observer (see [13] for details).

IS-MPC solves at each time instant t_k the following Quadratic Program (QP):

$$\begin{cases} \min\limits_{\boldsymbol{u}_k} \displaystyle\sum_{i=k}^{k+C-1} (\dot{x}_z^i)^2 + (\dot{y}_z^i)^2 + \alpha\big((x_z^{i+1} - x_{\text{mb}}^{i+1})^2 + \big(y_z^{i+1} - y_{\text{mb}}^{i+1}\big)^2\big) \\ \text{subject to:} \\ \bullet \text{ ZMP constraints} \\ \bullet \text{ stability constraints,} \end{cases}$$

where $\boldsymbol{u}_k = (\dot{x}_z^k, \ldots, \dot{x}_z^{k+C-1}, \dot{y}_z^k, \ldots, \dot{y}_z^{k+C-1})$ collects the decision variables over the control horizon, and α is a weight that modulates the relative importance of the second term whose role is to bring the ZMP close to the center of the moving box.

As customary in MPC, we extract the first sample $(\dot{x}_z^k, \dot{y}_z^k)$ from the QP solution and use it, together with the current disturbance estimate $(\hat{w}_x^k, \hat{w}_y^k)$, to integrate the dynamics (2). The resulting CoM position and velocity $(\boldsymbol{p}_c, \dot{\boldsymbol{p}}_c)$ are used as references $(\boldsymbol{p}_c^*, \dot{\boldsymbol{p}}_c^*)$ for the whole-body controller together with the relative hand positions and velocities $(\boldsymbol{r}_i^*, \dot{\boldsymbol{r}}_i^*)$, for $i = l, r$.

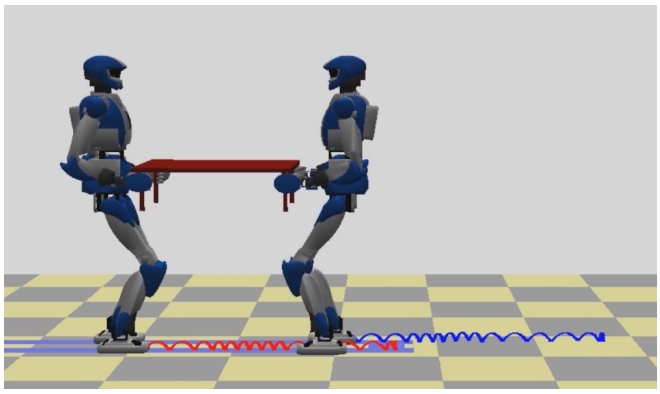

Fig. 4. Simulation 1: cooperative transportation of a table with the leader varying its speed. The red/blue lines are the right foot trajectories of leader/follower, while the footstep plan of the leader is shown in blue on the ground.

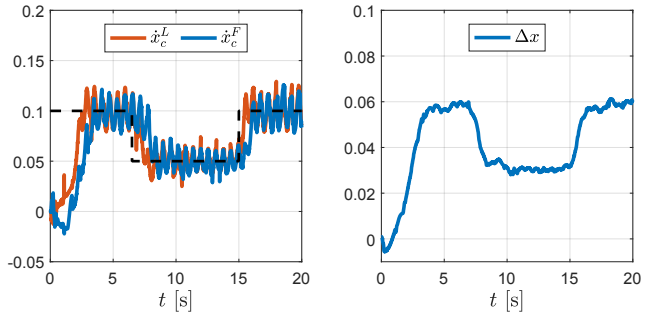

Fig. 5. Simulation 1. Left: x-component of the CoM velocity for the leader (red) and the follower, compared to the reference velocity of the object (dashed). Right: x-component of the nominal displacement $\Delta \boldsymbol{p}$.

VII. Results

We now present a selection of illustrative results from dynamic simulations performed in DART; see the video at `https://youtu.be/ChfhpSJFqg0` for animated clips. Both the leader and the follower are HRP-4 humanoids, while the manipulated objects are first a table and then a bar with two handles, both with a mass of 10 kg. Force measurements are obtained by querying the simulator for the contact forces at the hands[3]. The IS-MPC QP is solved using HPIPM, while the footstep planner QP uses IPOPT, interfaced from casADi. The simulation runs at a 100 Hz, and all computations are performed in real time.

Except when noted, the following parameters are used: $m = 38$ kg, $\bar{z}_c = 0.75$ m, $T_{\text{step}} = 0.6$ s, $F = 10$, $\delta = 0.01$ s, $C = 166$, $d_{z,x} = 0.07$ m, $d_{z,y} = 0.07$ m, $\alpha = 10^3$, $d_{a,x} = 0.4$ m, $d_{a,y} = 0.1$ m, $\ell = 0.2$ m, $k_{p,x} = 1$, $k_{d,x} = 0.1$, $k_{i,x} = 0.1$, $k_{p,y} = 1.2$, $k_{d,y} = 0.15$, $k_{i,y} = 0.05$, $\boldsymbol{M} = \text{diag}\{5, 5, 5\}$ kg. Different stiffness and damping are used for the hand admittance controller for the leader and the follower, respectively $\boldsymbol{K}^L = \text{diag}\{500, 500, 500\}$ N/m, $\boldsymbol{C}^L = \text{diag}\{200, 200, 200\}$ N·s/m and $\boldsymbol{K}^F = \text{diag}\{20, 20, 100\}$ N/m, $\boldsymbol{C}^F = \text{diag}\{50, 50, 150\}$ N·s/m.

[3]While this might seem as an ideal sensor simulation, forces measured in this way are actually very noisy, due the fact that contacts are broken and reestablished very frequently. For this reason, force measurements are preliminarly fed to a low-pass filter with a cutoff frequency of 3.18 Hz

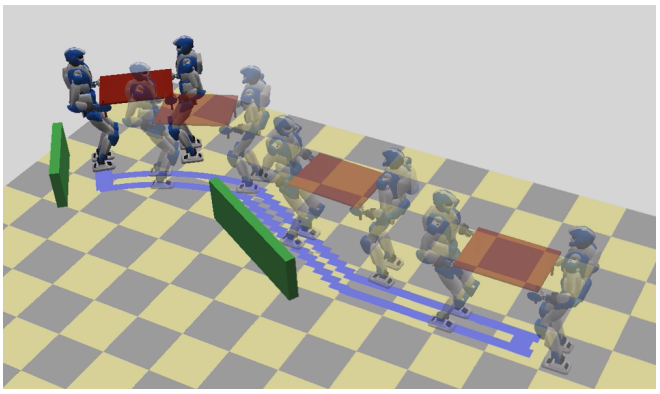

Fig. 6. Simulation 2 (stroboscopic view): cooperative transportation of a table with the leader moving along a complex path including rectilinear, diagonal, lateral and curved segments.

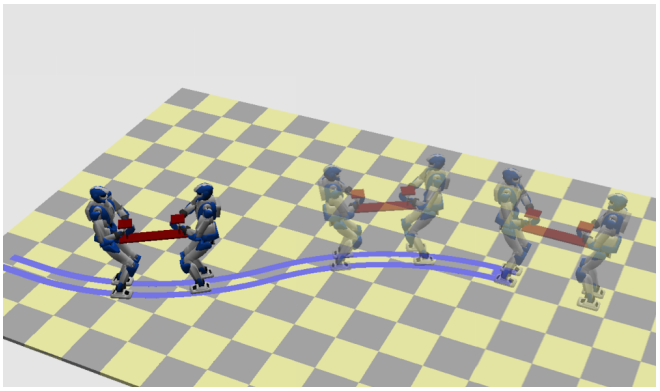

Fig. 7. Simulation 3 (stroboscopic view): cooperative transportation of a bar with the leader moving along a curved path.

In Simulation 1 (Fig. 5, left), the two robots walk in a straight line while transporting the table. The reference velocity of the leader is initially set to 0.05 m/s, then doubled, and then set back to the initial value. Changes in velocity of the leader are quickly accommodated by changes in the velocity of the follower, as shown in Fig. 5.

Simulation 2 (Fig. 6) is still a table transportation case, but with the leader now moving along a complex path involving rectilinear, lateral, diagonal, and curved segments in order to avoid certain obstacles in the workspace. Thanks to the proposed framework, the follower robot is able to adapt to the motion of the leader, and in particular to correctly infer its direction changes.

In Simulation 3 (Fig. 7) the robots must transport a long bar by using the handles at its extremities. This is a degenerate case for constructing the transportation axis, which must then be defined as in (Fig. 7), right. Again, the follower behaves satisfactorily and the transportation task is successfully completed.

Simulation 4 (Fig. 8) refers to a more challenging scenario. The robots are placed on opposite sides of a tall obstacle with variable height. The leader is instructed to walk sideways while lifting the object to avoid the protrusion in the middle. As before, the follower is able to accommodate the object

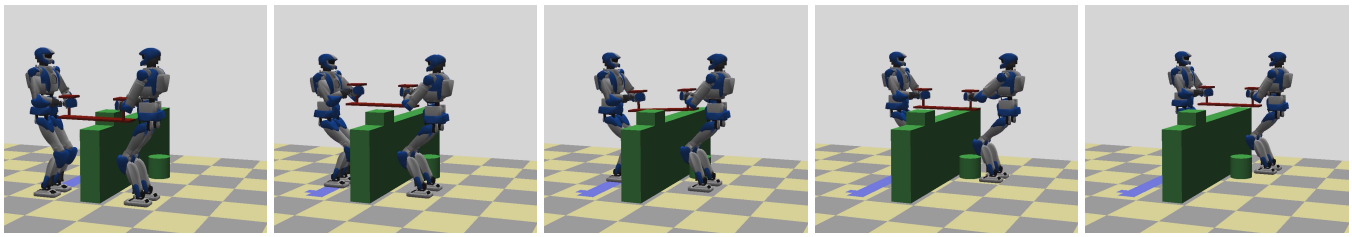

Fig. 8. Simulation 4: cooperative transportation of a bar over an obstacle. Note also the presence of a cylindrical obstacle further obstructing the motion of the follower.

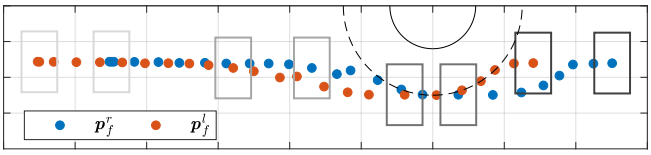

Fig. 9. Simulation 4: placement of the footsteps (blue for right, red for left) for the follower robot. In a few cases, the full footprint is shown to prove that there are no collisions. The boundary of the safe area $\mathcal{S}$ (dashed circle) is displaced with respect to the actual obstacle boundary (solid circle) to provide a safety margin.

motion; in addition, it also plan its footsteps so as to avoid the low cylindrical obstacle obstructing its motion, see Fig. 9. The stiffness of the hand admittance controller for the follower was lowered to 10 N/m in this case, to allow a quicker response to height changes.

VIII. Conclusions

We have presented a decentralized framework for cooperative transportation with humanoids. While the leader autonomously decides its motion based on the specification of the transportation task (and in fact can be replaced by a human), the follower infers the leader intentions (e.g, changes in speed or direction) from the perceived interaction forces and reacts accordingly to maintain its grasp on the object. The proposed framework includes a footstep planner capable of collision avoidance, but more advanced planners can be accommodated without modifications, e.g., for achieving 3D motion generation [18] or reaction to disturbances [19]).

Future work will include:

- using the vertical force measurement in place of its nominal value $\bar{f}^z$, thus obtaining a time-varying LIP to be used as a prediction model, see [20];
- allowing the leader to replan its footsteps by also taking into account the perceived motion of the follower, in order to handle critical situations, e.g., due to the follower being blocked by obstacles or — more in general — malfunctioning;
- performing experiments on a physical platform.

References

[1] I. Maroger, M. Silva, H. Pillet, N. Turpin, O. Stasse, and W. Bruno, "Walking paths during collaborative carriages do not follow the simple rules observed in the locomotion of single walking subjects," *Scientific Reports*, vol. 12, no. 1, p. 15585, 2022.

[2] M. J. Mataric, M. Nilsson, and K. T. Simsarin, "Cooperative multi-robot box-pushing," in *1995 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, vol. 3, 1995, pp. 556–561.

[3] C. R. Kube and H. Zhang, "Collective robotics: From social insects to robots," *Adaptive behavior*, vol. 2, no. 2, pp. 189–218, 1993.

[4] Z. Wang and M. Schwager, "Kinematic multi-robot manipulation with no communication using force feedback," in *2016 IEEE Int. Conf. on Robotics and Automation*. IEEE, 2016, pp. 427–432.

[5] D. Koung, O. Kermorgant, I. Fantoni, and L. Belouaer, "Cooperative multi-robot object transportation system based on hierarchical quadratic programming," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6466–6472, 2021.

[6] E. Tuci, M. H. Alkilabi, and O. Akanyeti, "Cooperative object transport in multi-robot systems: A review of the state-of-the-art," *Frontiers in Robotics and AI*, vol. 5, p. 59, 2018.

[7] L. Hawley and W. Suleiman, "Control framework for cooperative object transportation by two humanoid robots," *Robotics and Autonomous Systems*, vol. 115, pp. 1–16, May 2019.

[8] J. Lanini, H. Razavi, J. Urain, and A. Ijspeert, "Human intention detection as a multiclass classification problem: Application in physical human–robot interaction while walking," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4171–4178, 2018.

[9] D. J. Agravante, A. Cherubini, A. Sherikov, P.-B. Wieber, and A. Kheddar, "Human-humanoid collaborative carrying," *IEEE Trans. on Robotics*, vol. 35, no. 4, pp. 833–846, 2019.

[10] M.-H. Wu, A. Konno, and M. Uchiyama, "Cooperative object transportation by multiple humanoid robots," in *2011 IEEE/SICE Int. Symp. on System Integration*, 2011, pp. 779–784.

[11] A. Bussy, P. Gergondet, A. Kheddar, F. Keith, and A. Crosnier, "Proactive behavior of a humanoid robot in a haptic transportation task with a human partner," in *2012 IEEE RO-MAN: The 21st IEEE Int. Symp. on Robot and Human Interactive Communication*, 2012, pp. 962–967.

[12] N. Scianca, D. De Simone, L. Lanari, and G. Oriolo, "MPC for humanoid gait generation: Stability and feasibility," *IEEE Trans. on Robotics*, vol. 36, no. 4, pp. 1171–1188, 2020.

[13] F. M. Smaldone, N. Scianca, V. Modugno, L. Lanari, and G. Oriolo, "Gait generation using intrinsically stable MPC in the presence of persistent disturbances," in *2019 IEEE-RAS Int. Conf. on Humanoid Robots*, 2019, pp. 651–656.

[14] D. J. Agravante, A. Cherubini, A. Bussy, P. Gergondet, and A. Kheddar, "Collaborative human-humanoid carrying using vision and haptic sensing," in *2014 IEEE Int. Conf. on Robotics and Automation*, 2014, pp. 607–612.

[15] F. Flacco, A. Paolillo, and A. Kheddar, "Residual-based contacts estimation for humanoid robots," in *2016 IEEE-RAS Int. Conf. on Humanoid Robots*, 2016, pp. 409–415.

[16] A. Aboudonia, N. Scianca, D. De Simone, L. Lanari, and G. Oriolo, "Humanoid gait generation for walk-to locomotion using single-stage MPC," in *2017 IEEE-RAS Int. Conf. on Humanoid Robots*, 2017, pp. 178–183.

[17] T. Takenaka, T. Matsumoto, and T. Yoshiike, "Real time motion generation and control for biped robot - 1st report: Walking gait pattern generation-," in *2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 1084–1091.

[18] M. Cipriano, P. Ferrari, N. Scianca, L. Lanari, and G. Oriolo, "Humanoid motion generation in a world of stairs," *Robotics and Autonomous Systems*, vol. 168, p. 104495, 2023.

[19] M. Cipriano, M. R. Maximo, N. Scianca, L. Lanari, and G. Oriolo, "Feasibility-aware plan adaptation in humanoid gait generation," in *2023 IEEE-RAS Int. Conf. on Humanoid Robots*, 2023, pp. 1–8.

[20] F. Smaldone, N. Scianca, L. Lanari, and G. Oriolo, "From walking to running: 3D humanoid gait generation via MPC," *Frontiers in Robotics and AI*, vol. 9, 2022.