International Journal of Humanoid Robotics

PRIORITIZED MULTI-OBJECTIVE DYNAMICS AND CONTROL OF ROBOTS IN HUMAN ENVIRONMENTS

LUIS SENTIS AND OUSSAMA KHATIB
Artificial Intelligence Laboratory
Stanford University
Stanford, CA 94305, USA
{*lsentis, khatib*}*@robotics.stanford.edu*

Received (Day Month Year)
Revised (Day Month Year)
Accepted (Day Month Year)

We characterize the dynamics of multiple prioritized control objectives and we present a framework that can simultaneously control these objectives by providing decoupled closed-loop dynamics within their priority level. This controller is suitable for robots that operate in human environments, where they face multiple contacts and must comply with multiple control criteria. We explore several cases including constrained hand tracking and compliant multi-objective control.

Keywords: Prioritization, multi-objective dynamics, multi-objective control, Operational Space Formulation.

1. Introduction

A new generation of redundant robots that operate interactively in human environments [11] is emerging. Multi-arm robots, humanoids, and biomechanical subjects are a few examples. The control of these robots is challenging because we must control multiple objectives simultaneously and we must define a prioritized hierarchy to ensure conflict resolution.

Multi-objective controllers were first studied at the inverse kinematic level [2,4,13,14,18]. For higher performance, the *Operational Space Formulation* [7,8] was introduced to address the dynamic interactions between end-effector motions and forces. In the Operational Space Formulation, an operational task is controlled, and a dynamically consistent null-space describes the additional self-motion. Multiple operational tasks can be controlled if they are concatenated into a single task vector. Although this non-prioritized structure is straightforward, if a control conflict occurs, a tracking error will affect all the conflicting tasks.

We can solve this problem by controlling secondary tasks (a.k.a. postures) with the self-motion. The *Extended Operation Space Formulation* [15] supports dynamic control of the posture, but to this point it has only been applied to joint-space

postures. We presented previously [12] a broader extension to the Operational Space Formulation that characterizes and controls a secondary task or posture in the null-space of the main task. This approach can control postures defined in a variety of motion spaces; for example, it can control additional operational tasks or human-like postures, such as upright stand, effort minimization, or body self-balance.

As robot movement becomes more elaborate, we need to combine a growing number of control objectives. To ensure that critical objectives are fulfilled, we must establish a prioritized hierarchy, as well as a methodology to check the feasibility of the control objectives at a given level of their priority. We also need control strategies to deal with objectives that cannot be fully accomplished. In this paper, we describe precisely such a controller, and characterize the dynamic behavior of the prioritized objectives.

As an illustration of the potential and feasibility of this controller, we present two case studies of interactive whole-body control: (1) an example of interactive hand tracking under joint limit constraints, and (2) an example of compliant multi-objective control.

2. Task and Posture Control

We begin by describing the robot's dynamics in terms of its joint coordinates q,

$$A(q)\ddot{q} + b(q,\dot{q}) + g(q) = \Gamma. \tag{1}$$

For this system of n equations, Γ is the set of joint torques, $A(q)$ is the joint inertia matrix, $b(q,\dot{q})$ is the Coriolis and centrifugal torque vector, and $g(q)$ is the gravity torque vector.

The Operational Space Formulation provides a decomposition that describes the dynamics of a primary task and a posture that operates in the task-consistent null-space according to the torque equation

$$\Gamma = \Gamma_{task} + \Gamma_{posture}. \tag{2}$$

For an m dimensional operational task $x_t(q)$ with Jacobian $J_t(q) = \partial x_t(q)/\partial q$, the projection of the joint dynamics into the task space will render the $m \times m$ dynamic equation given by

$$\Lambda_t \ddot{x}_t + \mu_t + p_t = F_t, \tag{3}$$

where F_t is a force control vector in the task space, $\Lambda_t = \left(J_t A^{-1} J_t^T\right)^{-1}$ is the task inertia matrix, μ_t is the Coriolis and centrifugal force vector, and p_t is the gravity force vector. The control input

$$\Gamma_{task} = J_t^T F_t, \tag{4}$$

$$F_t = \Lambda_t \ddot{x}_{ref(t)} + \mu_t + p_t, \tag{5}$$

provides the decoupled dynamic behavior $\ddot{x}_t = \ddot{x}_{ref(t)}$, where $\ddot{x}_{ref(t)}$ is an acceleration reference input. The task-consistent null-space [6] is defined by

$$N_t(q) = I - \overline{J}_t J_t, \tag{6}$$

where $\overline{J}_t = A^{-1} J_t^T \Lambda_t$ is the dynamically-consistent generalized inverse of J_t.

A posture is defined as a linear combination of the columns of N_t^T according to

$$\Gamma_{posture} = N_t^T \Gamma_{null}. \tag{7}$$

It can be used to control a secondary task $x_p(q)$ with Jacobian $J_p = \partial x_p / \partial q$. In a previous paper, [12] we introduced a new kinematic space defined by the Jacobian $J_{p|t} = J_p N_t$, resulting from projecting the joint rates into the task consistent null-space. An l dimensional posture is characterized by the $l \times l$ inertia matrix $\Lambda_{p|t} = \left(J_{p|t} A^{-1} J_{p|t}^T\right)^{-1}$. Its behavior is described by the dynamic equation

$$\Lambda_{p|t} \ddot{x}_{p|t} + \mu_{p|t} + p_{p|t} = F_{p|t}, \tag{8}$$

where $F_{p|t}$ is the control force vector, and $\mu_{p|t}$ and $p_{p|t}$ are, respectively, the Coriolis/centrifugal, and gravity force vectors of the posture. We can accomplish dynamically-consistent control of posture [9] by choosing the control input

$$\Gamma_{posture} = J_{p|t}^T F_{p|t}, \tag{9}$$

$$F_{p|t} = \Lambda_{p|t}\left(\ddot{x}_{ref(p)} - \ddot{x}_{p|bias}\right) + \mu_{p|t} + p_{p|t}, \tag{10}$$

where $\ddot{x}_{ref(p)}$ is a posture-control reference. Here, $\ddot{x}_{p|bias}$ is a bias acceleration induced by the coupling of the primary task into the posture; it includes corrections for Coriolis/centrifugal and gravity forces. Its decomposition is given by the equation

$$\ddot{x}_{p|bias} = J_p A^{-1} \Gamma_{task} + \Lambda_{p|t}^{-1}\left(\mu_{p|t} + p_{p|t}\right) - \Lambda_p^{-1}\left(\mu_p + p_p\right). \tag{11}$$

If $J_{p|t}$ is full rank, the posture is fully controllable and this controller will yield the decoupled dynamic behavior $\ddot{x}_p = \ddot{x}_{ref(p)}$. Figure 1 illustrates the control of an upright posture with simultaneous control of the position of the hands and the robot's self-balance.

A posture is partially controllable when the Jacobian $J_{p|t}$ drops rank. The inertial properties of the posture approach infinity in the uncontrollable directions. Under these circumstances, the inertia matrix can be expressed by the following eigen-decomposition

$$\Lambda_{p|t}^{-1} = J_{p|t} A^{-1} J_{p|t}^T = \begin{bmatrix} U_r & U_n \end{bmatrix} \begin{bmatrix} \Sigma_r & \\ & 0_{(l-k)\times(l-k)} \end{bmatrix} \begin{bmatrix} U_r^T \\ U_n^T \end{bmatrix}, \tag{12}$$

where k represents the number of uncontrollable directions, Σ_r is a $k \times k$ diagonal matrix of non-zero eigenvalues (r stands for rank), U_r is an $l \times k$ matrix with columns corresponding to the non-zero eigenvectors, and U_n is an $l \times (l-k)$ matrix corresponding to the zero eigenvectors (n stands for null). Because some eigenvalues are equal to zero, it is not possible to obtain full decoupling of $\ddot{x}_p$, however, if the control input is chosen to be

$$F_{p|t} = \left(U_r \Sigma_r^{-1} U_r^T\right)\left(\ddot{x}_{ref(p)} - \ddot{x}_{p|bias}\right) + \mu_{p|t} + p_{p|t}, \tag{13}$$

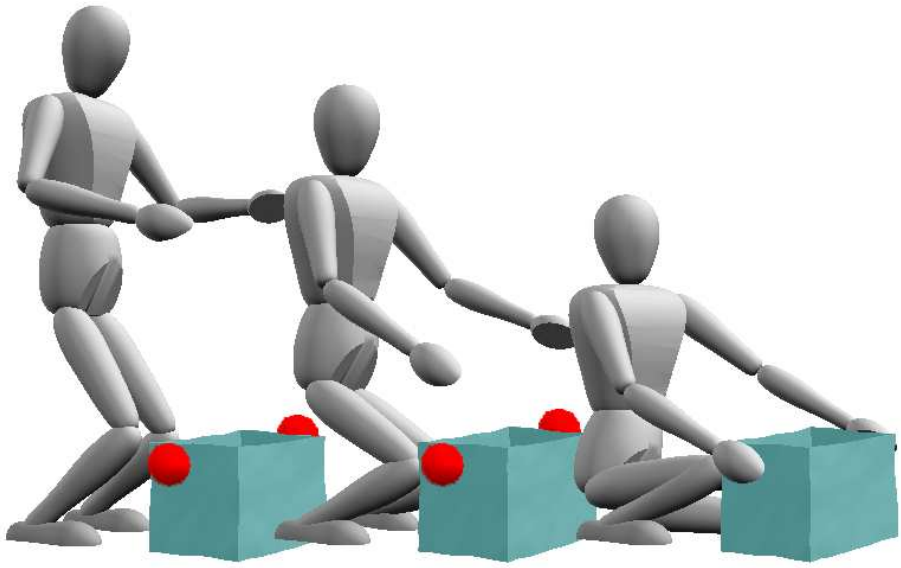

Fig. 1. **Task and Posture Decomposition:** In this sequence, the robot's task is to reach the sides of the box while maintaining self-balance and upright posture. The spheres represent the target positions of the robot's hands.

we will accomplish dynamic decoupling in the controllable directions U_r according to $U_r^T\big(\ddot{x}_p = \ddot{x}_{ref(p)}\big)$.

To control whole-body movements effectively, we must ensure that multiple objectives are accomplished. These objectives comprise a collection of operational tasks, physical constraints, and body postures. In the following section we present an extension that allows us to introduce a multi-level prioritized hierarchy.

3. Dynamic Control of Multiple Objectives

As interactive applications of redundant robots gain importance, robots are required to control multiple objectives simultaneously to accomplish a global task effectively. In particular, we require a hierarchy to ensure the fulfillment of critical objectives, such as physical constraints and body self-balance, while optimizing the execution of interactive tasks. In this section, we propose an extension to the task and posture decomposition previously described; this extension allows us to characterize the dynamic behavior of multiple objectives and to control those objectives according to pre-set priorities.

We assume that, to accomplish a global task, the robot must control simultaneously a collection of N objectives, $\{x_k(q)\,|\,k = 1, 2, \ldots, N\}$ with Jacobians $J_k(q) = \partial x_k(q)/\partial q$. These objectives comprise physical constraints, manipulation

and walking tasks, and body postures. The whole-body control torque Γ is given by

$$\Gamma = \sum_{k=1}^{N} \Gamma_{obj(k)}, \tag{14}$$

where $\Gamma_{obj(k)}$ are the torques to control the objective K. We wish to assign priorities to every objective. For example, physical constraints could take highest priority, followed by manipulation tasks, with body postures the lowest priority. Let us assume that the objectives are numbered according to the desired hierarchy.

Similar to our use of Equation 7, we can establish the hierarchy through a linear combination of null-space columns according to

$$\Gamma_{obj(k)} = N_{P(k)}^T \Gamma_{null(k)}\,, \tag{15}$$

where $P(k) = \{1, \cdots, k-1\}$ represents the set of *preceding* objectives, $N_{P(k)}$ is the null-space of *all* preceding objectives $P(k)$, and $\Gamma_{null(k)}$ is the control input in the null-space. Based on the task/posture decomposition of Equations 4 and 9, we express the following relationship between torque and force

$$\Gamma_{obj(k)} = J_{k|P(k)}^T F_{k|P(k)}, \tag{16}$$

where the subscript $k|P(k)$ indicates that the objective k is controlled in the null-space of $P(k)$, and

$$J_{k|P(k)} \triangleq J_k N_{P(k)}. \tag{17}$$

This Jacobian results from projecting the joint velocities into the null-space. We can also associate a prioritized inertia matrix defined by

$$\Lambda_{k|P(k)} = \left(J_{k|P(k)} A^{-1} J_{k|P(k)}^T\right)^{-1}. \tag{18}$$

3.1. *Prioritized Null Space*

We can establish a dynamic hierarchy by choosing a null-space with no acceleration effects in all preceding objectives. We do so by defining the constraint

$$\forall\, i \in P(k) \quad J_i A^{-1} N_{P(k)}^T = 0. \tag{19}$$

The null-space that fulfills the preceding constraint has the following unique solution

$$N_{P(k)} = I - \sum_{i=1}^{k-1} \overline{J}_{i|P(i)} J_{i|P(i)}, \tag{20}$$

where $\overline{J}_{i|P(i)} = A^{-1} J_{i|P(i)}^T \Lambda_{i|P(i)}$ is the priority-consistent generalized inverse of $J_{i|P(i)}$.

Proof by Induction of Equation 20:

(1) For $k = 2$, $J_1 A^{-1} N_{P(2)}^T = J_1 A^{-1} - J_1 A^{-1} J_i^T \Lambda_1 J_1 A^{-1} = 0$.
(2) For any k and $\forall i \in P(k)$ let us assume $J_i A^{-1} N_{P(k)}^T = 0$.
(3) For $k+1$, $\forall i \in P(k+1)$, and using (2),

$$J_i A^{-1} N_{P(k+1)}^T = J_i A^{-1} N_{P(k)}^T \big(I - J_{k|P(k)}^T \overline{J}_{k|P(k)}^T\big) = 0. \qquad \square$$

Here, we have used the properties: $\forall k \ \big(N_{P(k)}\big)^2 = N_{P(k)}$, and $J_k A^{-1} N_{P(k)}^T = 0$.

3.2. *Control of Prioritized Objectives*

The dynamic behavior of a prioritized objective k can be obtained by projecting the joint dynamics according to

$$\overline{J}_{k|P(k)}^T \Big(A\ddot{q} + b + g = \Gamma_{obj(k)} \Big) \Longrightarrow$$

$$\Lambda_{k|P(k)} \ddot{x}_{k|P(k)} + \mu_{k|P(k)} + p_{k|P(k)} = F_{k|P(k)}, \quad (21)$$

where $\mu_{k|P(k)}$ and $p_{k|P(k)}$ are the Coriolis/centrifugal and gravity force vectors of the prioritized objective. We accomplish efficient control of the objective $x_k(q)$ by choosing the control torque

$$\Gamma_{obj(k)} = J_{k|P(k)}^T F_{k|P(k)}, \quad (22)$$

$$F_{k|P(k)} = \Lambda_{k|P(k)} \big(\ddot{x}_{ref(k)} - \ddot{x}_{k|bias} \big) + \mu_{k|P(k)} + p_{k|P(k)}. \quad (23)$$

Here $\ddot{x}_{k|bias}$ is a bias acceleration induced by the coupling of preceding objectives; it also includes a correction for the Coriolis/centrifugal and gravity forces according to

$$\ddot{x}_{k|bias} = J_k A^{-1} \sum_{i=1}^{k-1} \Gamma_{obj(i)} + \Lambda_{k|P(k)}^{-1} \big(\mu_{k|P(k)} + p_{k|P(k)} \big) - \Lambda_k^{-1} \big(\mu_k + p_k \big). \quad (24)$$

If $J_{k|P(k)}$ is full rank, this controller will yield the decoupled dynamic behavior $\ddot{x}_k = \ddot{x}_{ref(k)}$, where $\ddot{x}_{ref(k)}$ is the objective control reference.

3.3. *Controllability of Objectives*

Similarly to our definition of posture controllability (see Equations 12 and 13), we say that an objective k is partially controllable in the hierarchy if the Jacobian $J_{k|P(k)}$ drops rank. In particular, the inertia matrix has the following eigendecomposition

$$\Lambda_{k|P(k)}^{-1} = J_{k|P(k)} A^{-1} J_{k|P(k)}^T = \begin{bmatrix} U_{r(k)} \ U_{n(k)} \end{bmatrix} \begin{bmatrix} \Sigma_{r(k)} & \\ & 0 \end{bmatrix} \begin{bmatrix} U_{r(k)}^T \\ U_{n(k)}^T \end{bmatrix}, \quad (25)$$

where $\Sigma_{r(k)}$ is a diagonal matrix of non-zero eigenvalues, $U_{r(k)}$ is a matrix corresponding to non-zero eigenvectors, and $U_{n(k)}$ is a matrix corresponding to zero

eigenvectors. Because of the zero eigenvalues, it is not possible to control $\ddot{x}_k$ fully. However, by choosing the control input

$$F_{k|P(k)} = \left(U_{r(k)}\Sigma^{-1}_{r(k)}U^T_{r(k)}\right)\left(\ddot{x}_{ref(k)} - \ddot{x}_{k|bias}\right) + \mu_{k|P(k)} + p_{k|P(k)}, \qquad (26)$$

we accomplish dynamic decoupling in the controllable directions according to $U^T_{r(k)}\left(\ddot{x}_k = \ddot{x}_{ref(k)}\right)$, where $U_{r(k)}$ defines the controllable modes.

4. Simulation Environment

To verify the proposed controller, we developed a humanoid robotic model that can be simulated and controlled in SAI [10]. SAI is a unique virtual environment that integrates multi-body dynamics [1], robot control, multi-contact simulation, and haptic interaction. It incorporates a dynamics engine that resolves forward and inverse dynamics of an n degrees-of-freedom (DOF) branching multi-body system with linear complexity, $O(n)$. Moreover, we can resolve p collisions with a complexity of $O(np + p^3)$ using operational space models [16]. Figure 2 displays a sequence of snapshots from simulated falling of our humanoid.

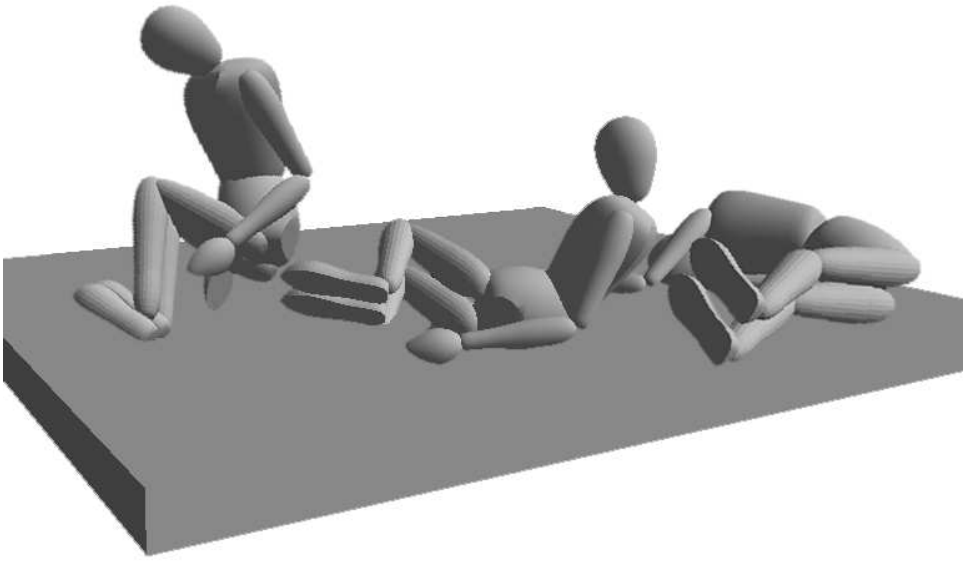

Fig. 2. **Robot Simulation:** This SAI simulation shows a robot falling due to gravity. Efficient multi-body dynamics and multi-contact algorithms resolve the simulation in real-time.

5. Case Study: Interactive Hand Control with Joint Limit Constraints

To study the proposed controller, we first explore an example of interactive hand control with joint-limit constraints. Our humanoid robot model consists of $n = 24$ DOF: 2×6 for the legs, 2×4 for the arms, 2 for the torso, and 2 for head. The robot's height is $1.65m$; its weight is $71Kg$. In this example, the robot is commanded to

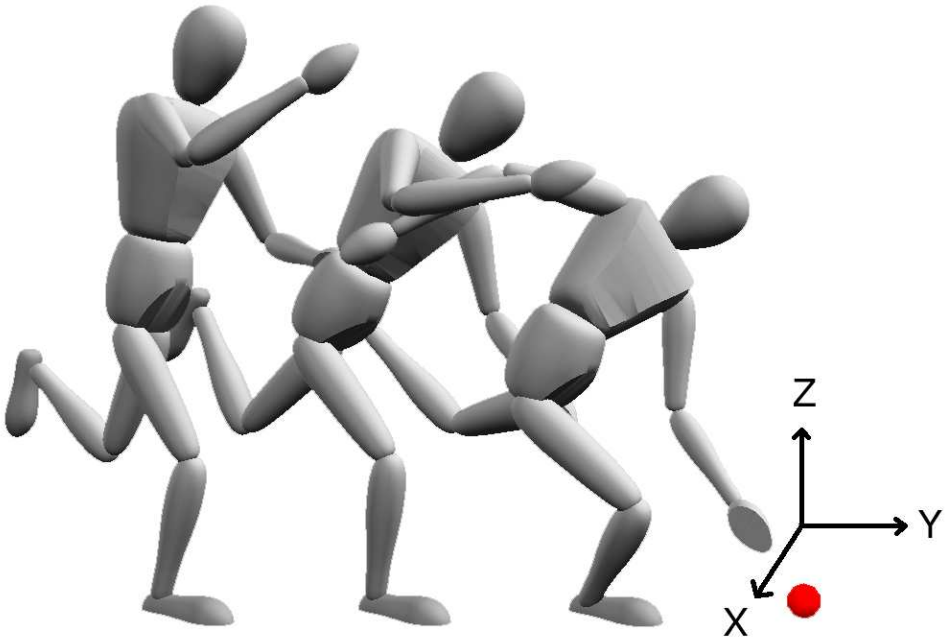

Fig. 3. **Hand Control with Joint-Limit Constraints:** In this sequence, the robot is commanded to reach a target with its left hand while maintaining self-balance and without violating joint-limit constraints. In the rightmost snapshot, the robot cannot reach farther due to joint limits in the left elbow, right-leg, and upper body.

reach a target point with its left hand. To achieve this task, the robot must control four objectives: joint-limit constraints, self-balance, hand control, and whole-body posture. Self-balance is based on the control of the global center of mass according to

$$x_{com} = \frac{1}{M}\sum_{i=1}^{n} m_i x_{com(i)} \qquad J_{com} = \frac{1}{M}\sum_{i=1}^{n} m_i J_{com(i)}, \tag{27}$$

where $x_{com(i)}$ represents the center of mass of link i and M is the robot's total mass. The control of the hand is based on the latter's Cartesian position x_{hand}. The posture maximizes joint manipulability. To accomplish the desired objectives

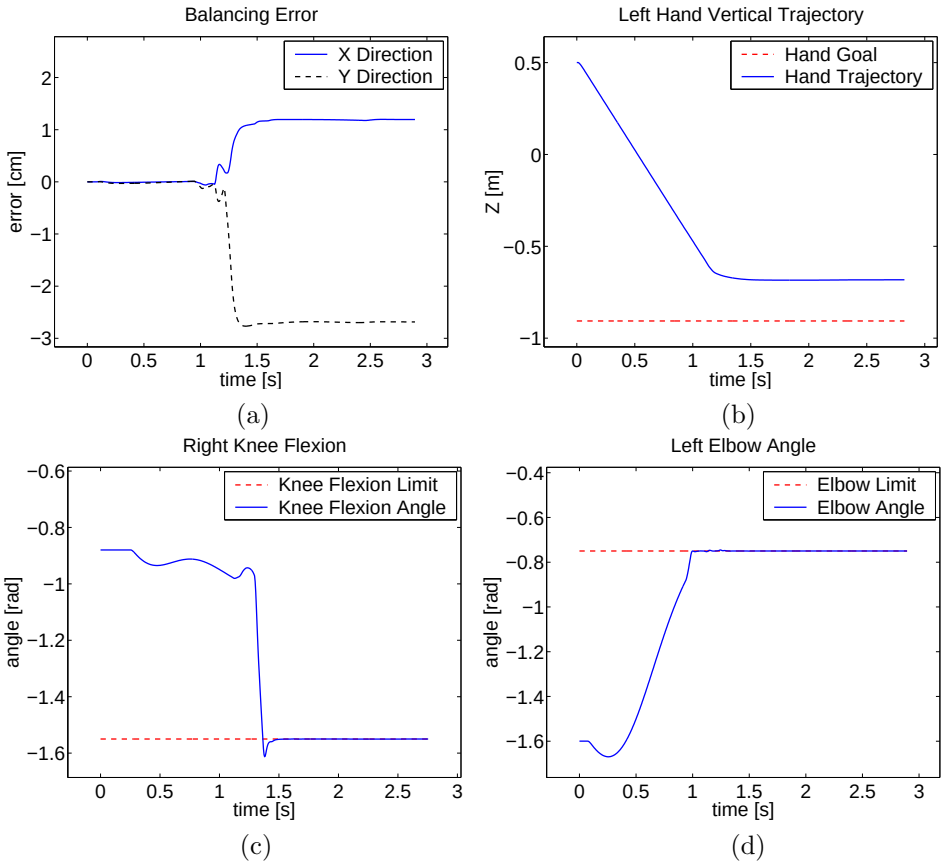

Fig. 4. **Data Recorded When Self-Balance and Hand Control Share Priority:** When the knee flexion and left elbow-joint limits are reached (c and d), the self-balance error increases substantially (a). The hand's vertical position is not permitted to reach its goal, due to joint limits (d).

we define the following *potential* functions [5]:

$$V_{JLC} = \| \, q_{violating} - q_{limit} \, \|^2 \qquad V_{HAND} = \| \, x_{hand} - x_{target} \, \|^2, \tag{28}$$

$$V_{BLN} = \| \, x_{com} - x_{foot} \, \|^2 \qquad V_{MNP} = \| \, W\big(q - q_{mid}\big) \, \|^2 \, . \tag{29}$$

The abbreviations JLC, BLN, and MNP stand for joint-limit constraints, self-balance, and joint manipulability respectively. In addition, $q_{violating}$ is the vector of robot joints that, at a given time, violate joint limits, q_{limit} is the vector of joint-limit values, x_{target} is an interactive hand target, x_{foot} is the position of the right foot, $q_{mid} = (q^-_{JL} + q^+_{JL})/2$ comprises the joint mid-range positions, $W = diag(\, q^+_{JL} - q^-_{JL} \,)$ is a normalizing matrix, and q^+_{JL} and q^-_{JL} are the upper and lower joint limits. In

this experiment, we used the following control reference with velocity saturation

$$\ddot{x}_{ref(k)} = -k_v\big(\dot{x}_k - \nu\dot{x}_{des(k)}\big), \tag{30}$$

$$\dot{x}_{des(k)} = \frac{k_p}{k_v}\nabla V_k \qquad \nu = min\bigg(1, \frac{v_{max(k)}}{||\dot{x}_{des(k)}||}\bigg), \tag{31}$$

where $\dot{x}_{des(k)}$ is a desired *velocity* and $v_{max(k)}$ is a saturation value.

To illustrate the advantages of prioritizing the objectives, we first study a controller that sets equal priorities for the self-balance and hand objectives. This condition can be expressed by the following torque equation

$$\Gamma = \Gamma_{JLC}(1) + \Gamma_{BLN}(2) + \Gamma_{HAND}(2) + \Gamma_{MNP}(3), \tag{32}$$

where the priorities are displayed in parentheses. We control the self-balance and hand objectives by combining them into a joined objective with Jacobian matrix and control input defined by

$$J_{combined} = \begin{bmatrix} J_{BLN} \\ J_{HAND} \end{bmatrix} \qquad \ddot{x}_{ref(combined)} = \begin{bmatrix} \ddot{x}_{ref(BLN)} \\ \ddot{x}_{ref(HAND)} \end{bmatrix}. \tag{33}$$

When we apply the controller defined in Equation 23 we obtain the results shown in Figure 4. The error in self-balance is zero while the hand moves down with steady speed. When the hip, elbow, and knee flexion joint limits are reached at $t = 0.9s$, $1s$, and $1.2s$ respectively, self-balance and hand control cannot be accomplished simultaneously. Because these two objectives have equal priority, an error will appear according to their control gains. The steady-state error associated with self-balance is $1cm$ in the X direction and $3cm$ in the Y direction, while the hand stops approximately $22cm$ away from its target due to joint limits.

Next, we assign a higher priority to the self-balance control objective, according to

$$\Gamma = \Gamma_{JLC}(1) + \Gamma_{BLN}(2) + \Gamma_{HAND}(3) + \Gamma_{MNP}(4). \tag{34}$$

We observe that even though the hip, elbow, and knee flexion joint limits are reached, the maximum error in self-balance is only $2mm$ in the X and Y directions. The hand is still unable to reach its goal, also due to joint limits. Instead, it reaches the closest possible position $24cm$ away from its target, $2cm$ farther away than it reached the previous example. This difference is expected, because self-balance is now fully accomplished.

In conclusion, critical objectives need to be assigned the highest priority to avoid conflicts in which these objectives may be compromised by less important ones.

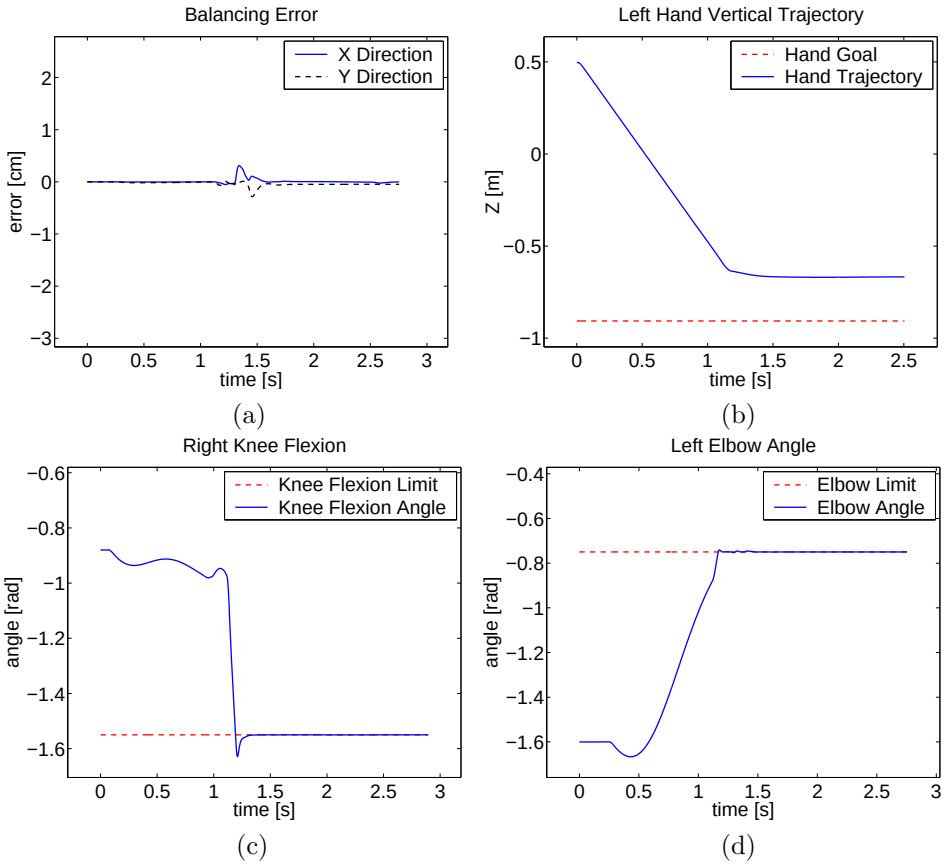

Fig. 5. **Data Recorded When Self-Balance Precedes Hand Control:** The self-balance error (a) stays small when the knee flexion and left elbow joint limits are reached (c and d). Because the hierarchy assigns it a higher priority, self-balance (a) remains undisturbed by hand control (b).

6. Case Study: Compliant Objectives

One of the advantages of a controller that provides decoupled closed loop dynamics at all levels is that the control gains can be greatly reduced without the tracking error being degraded. We can use this ability to add compliance to selected objectives. We shall study an example in which we combine four objectives, self-balance, left-hand control, upright posture, and joint manipulability, according to the torque equation

$$\Gamma = \Gamma_{BLN}(1) + \Gamma_{HAND}(2) + \Gamma_{UPR}(3) + \Gamma_{MNP}(4), \tag{35}$$

where the new subscript UPR stands for upright posture, and the number in parenthesis defines the arrangement in the control hierarchy. The self-balance, hand-control, and joint-manipulability objectives are controlled in the same manner as they where in Section 5. The upright posture consists of aligning the chest's Z axis

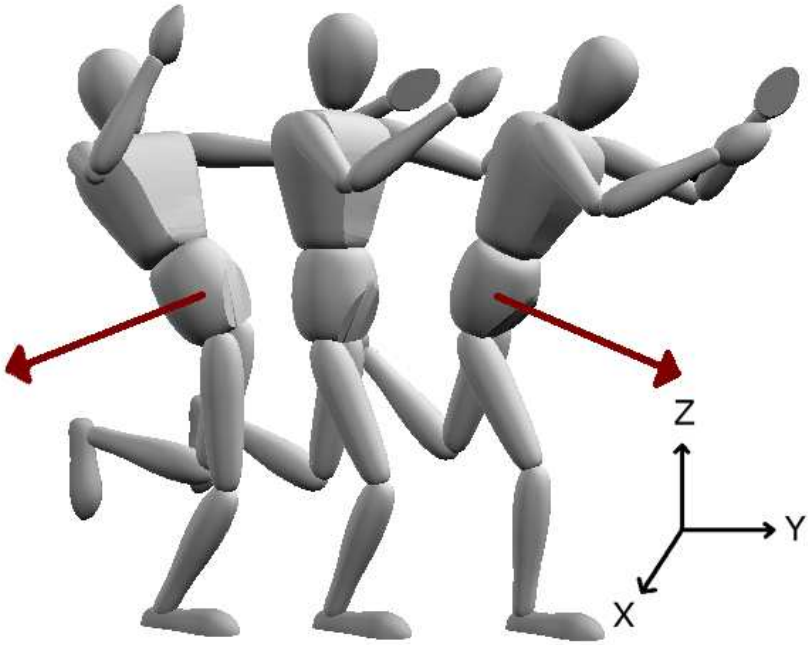

Fig. 6. **Compliant Upright Posture:** This sequence of snapshots shows an external force applied to the robot while the latter maintains self-balance, left-hand control, and upright posture. The upright-posture gains are low to provide compliance. Notice that self-balance and hand position are unaffected.

(vertical direction) with a desired upright orientation. The orientation of the chest is represented by the quaternion $\lambda_{r(chest)}$; the desired upright orientation is represented by the quaternion $\lambda_{r(upright)}$. We also consider the $3 \times n$ angular Jacobian of the chest $J_{r(chest)}$. The Jacobian and the control reference of the upright posture can be expressed as

$$J_{UPR} = S_z J_{r(chest)} \qquad \ddot{x}_{ref(UPR)} = -S_z(k_p \triangle\phi_{chest} + k_v \dot{\phi}_{chest}), \tag{36}$$

where S_z is a selection matrix that chooses the chest vertical direction, $\triangle\phi_{chest} = E_r(\lambda_{r(chest)} - \lambda_{r(upright)})$ is the posture angular error, E_r is a representation transformation, and $\dot{\phi}_{chest} = J_{r(chest)}\dot{q}$ is the instantaneous angular velocity of the chest.

We first study an example in which we allow low upright-posture gains while maintaining high self-balance and hand control gains. In particular, we choose the following gains: $k_{p(BLN)} = 5000$, $K_{p(HAND)} = 1000$ and $k_{p(UPR)} = 50$ with a servo rate of $1kHz$. As shown in Figure 6, we apply an external force to the hip of the robot in the Y direction. The shape of this force is sinusoidal; its amplitude is $500Nm$. The results are displayed in Figure 7.

The upright posture oscillates approximately $\pm 40^\circ$ due to the compliance in-

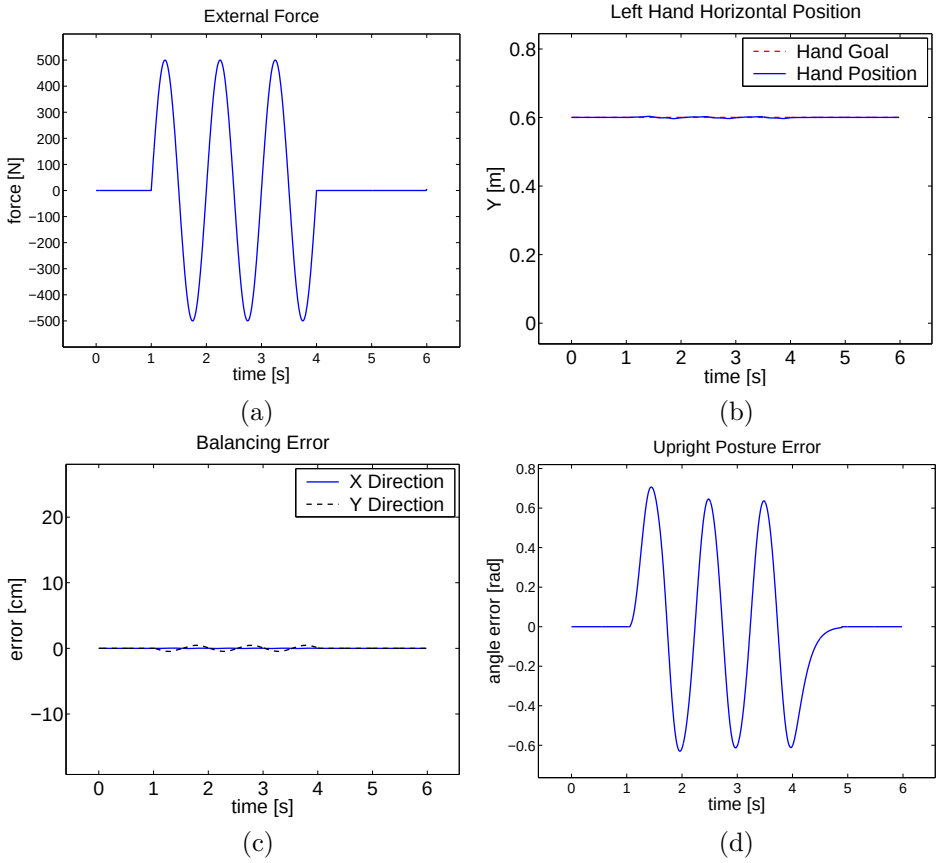

Fig. 7. **Data on Compliant Upright Posture:** When we apply an external force to the robot's hip (a), we induce a large oscillation to the upright posture (d) due to that objective's low control gain. In contrast, the self-balance (c) and hand-position errors (b) are small due to the high gains of these objectives.

duced by the low gain. In contrast, the self-balance error oscillates $\pm 0.5cm$, and the hand error oscillates $\pm 3.2mm$. Notice, in Figure 6, that the global center of gravity of the robot stays centered above the right foot despite the large forces applied to the hip. The left hand position does not move either, due to the decoupled dynamics and high gain.

We can look at a second example by applying a low self-balance gain while maintaining high gains in the hand control and upright posture. The gains are now changed to: $k_{p(BLN)} = 120$, $K_{p(HAND)} = 1000$ and $k_{p(UPR)} = 5000$. In Figure 8, we show snapshots of this experiment, with application of external force the same as in the previous example. The results are displayed in Figure 9. Observe that the self-balance task oscillates $\pm 15cm$ approximately, whereas the upright posture oscillates only $\pm 0.6°$. The hand error oscillates $\pm 3.5mm$. Figure 8 shows that the upright

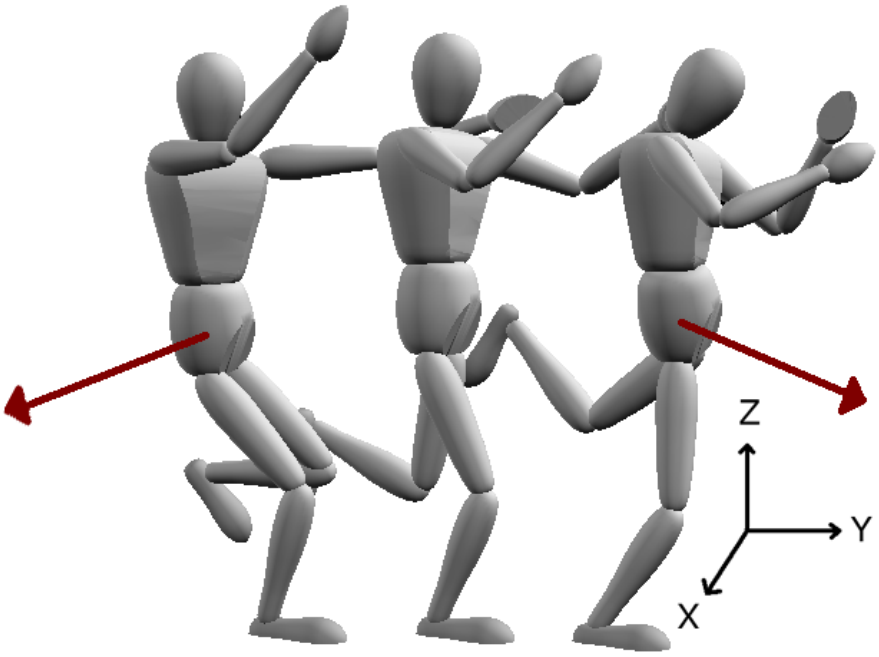

Fig. 8. **Compliant Self-Balance:** The same force as that in Figure 6 is applied while the robot maintains self-balance, left-hand control, and upright posture. The control gain in the self-balance task is now very low. The upright posture and hand position are undisturbed.

posture remains unchanged. Notice that the torso twists to the sides; this movement is a desired effect that results from controlling only the vertical orientation of the chest.

In conclusion, by adjusting the gains we can control the compliance of selected objectives.

7. Conclusion

To become practical, humanoid robots must be controlled interactively. However, most control approaches are based on offline trajectory planning. In contrast, our approach, addresses online control of global-tasks. In this paper, we proposed to break these tasks down into simpler control primitives, a.k.a. control objectives, and we presented a methodology to check for individual objective feasibility at run-time. This allows to change interactively individual objective parameters, such as gains and target positions, without the need to check in advance their feasibility. Furthermore, it allows to add and remove objectives at run-time without replanning.

Our controller characterizes for the first time the mass properties of multiple control objectives including null-space projections. To address hybrid position/force control we implemented operational-space dynamic controllers for each individual

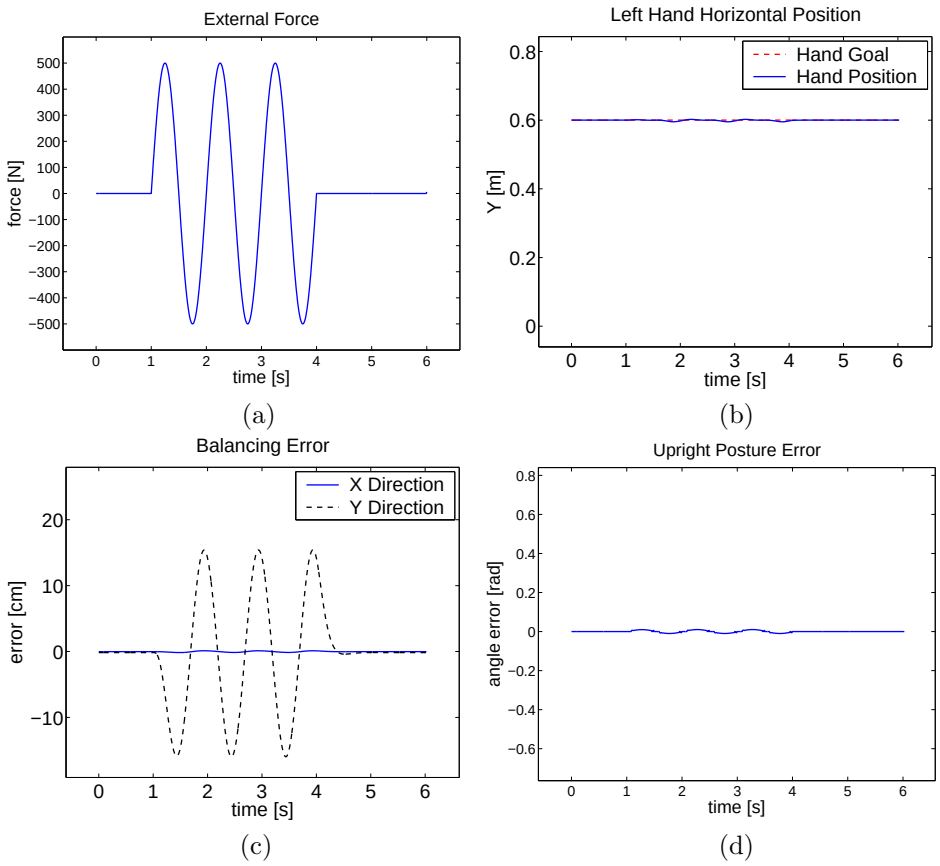

Fig. 9. **Data on Compliant Self-Balance:** An external force is applied to the robot's hip (a). A large oscillation is induced in the self-balance task (c), due to the latter's low gain. The hand-position and upright-posture errors (b and d) remain close to zero, due to the associated high gains.

objective. This allows us to design compliant/force control at the same level as the control of the objective position.

We envision humanoid robots in contact with multiple body parts at once. Our framework allows to individually control each contact point, while accomplishing other objectives such as manipulation, locomotion, and body postures.

While today the interactive control of humanoids is limited to the online selection of a few preplanned motions, with this new controller, we can construct complex behaviors by adding new control objectives, or by changing the individual control parameters.

Our next step is to apply hybrid position/force controllers to multiple body parts in contact. Another goal is to generalize walking sequences for this controller.

There is a fair amount of work on similar controllers at the inverse-kinematic

level. However, because we characterize the dynamics of individual objectives, our controller is suitable for high performance position/force control. Furthermore by controlling the compliance of each objective, we can provide robot safety while carrying out essential tasks such as self-balance, walking, or object manipulation.

Acknowledgments

We are grateful for the contributions of Jae-Heung Park, James Warren, Vincent De Sapio, Tine Lefebvre, François Conti, Irena Pashchenko, Kyong-Sok Chang, Diego Ruspini, and Oliver Brock. Our work was supported by the Honda humanoid project [3,17].

References

1. K.C. Chang and O. Khatib. Operational space dynamics: Efficient algorithms for modeling and control of branching mechanisms. In *Proceedings of the IEEE International Conference on Robotics and Automation*, April 2000.
2. H. Hanafusa, T. Yoshikawa, and Y. Nakamura. Analysis and control of articulated robot with redundancy. In *Proceedings of IFAC Symposium on Robot Control*, volume 4, pages 1927–1932, 1981.
3. K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka. The development of Honda humanoid robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 1321–1326, Leuven, Belgium, 1998.
4. J. M. Hollerbach and K. C. Suh. Redundancy resolution of manipulators through torque optimization. *International Journal of Robotics and Automation*, 3(4):308–316, 1987.
5. O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5(1):90–8, 1986.
6. O. Khatib. A Unified Approach to Motion and Force Control of Robot Manipulators: The Operational Space Formulation. *International Journal of Robotics and Automation*, RA–3(1):43–53, February 1987.
7. O. Khatib. A united approach to motion and force control of robot manipulators: The operational space formulation. *International Journal of Robotics Research*, 3(1):43–53, 1987.
8. O. Khatib. Object manipulation in a multi-effector robot system. In R. Bolles and B. Roth, editors, *Robotics Research 4*, pages 137–144. MIT Press, 1988.
9. O. Khatib. Inertial Properties in Robotics Manipulation: An Object-Level Framework. *International Journal of Robotics Research*, 14(1):19–36, 1995.
10. O. Khatib, O. Brock, K.C. Chang, F. Conti, D. Ruspini, and L. Sentis. Robotics and interactive simulation. *Communications of the ACM*, 45(3):46–51, March 2002.
11. O. Khatib, O. Brock, K.C. Chang, D. Ruspini, L. Sentis, and S. Viji. Human-centered robotics and interactive haptic simulation. *International Journal of Robotics Research*, 23(2), February 2004.
12. O. Khatib, L. Sentis, J.H. Park, and J. Warren. Whole body dynamic behavior and control of human-like robots. *International Journal of Humanoid Robotics*, 1(1):29–43, March 2004.
13. A.A. Maciejewski and C.A. Klein. Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. *International Journal of Robotics Research*, 4(3):109–117, 1985.

14. Y. Nakamura, H. Hanafusa, and T. Yoshikawa. Task-priority based control of robot manipulators. *International Journal of Robotics Research*, 6(2):3–15, 1987.
15. K. C. Park, P. H. Chang, and S. Lee. Analysis and control of redundant manipulator dynamics based on an extended operational space. *Robotica*, 19:649–662, 2001.
16. D. Ruspini and O. Khatib. Collision/Contact Models for Dynamic Simulation and Haptic Interaction. In *The 9th International Symposium of Robotics Research (ISRR'99)*, pages 185–195, Snowbird, USA, October 1999.
17. Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura. The intelligent ASIMO: System overview and integration. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2478–2483, Laussane, Switzerland, October 2002.
18. B. Siciliano and J. Slotine. A general framework for managing multiple tasks in highly redundant robotic systems. In *Proceedings of the IEEE International Conference on Advanced Robotics*, pages 1211–1216, Pisa, Italy, June 1991.

Luis Sentis is a doctoral candidate at the Stanford Artificial Intelligence Laboratory. He earned his B.S. degree from the Technical University of Catalonia and his M.S. from Stanford University, both in Electrical Engineering. His research emphasizes interactive robot control for human environments.

Oussama Khatib is a Professor of Computer Science and Mechanical Engineering at the Stanford Artificial Intelligence Laboratory. His research interests include humanoid robot control, biomechanical simulations, and haptics.