2012 9th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)
Daejeon, Korea / November 26-29, 2012

A Continuous Task Transition Algorithm for Operational Space Control Framework

Hyejin Han*, Jaemin Lee*** and Jaeheung Park*,**†
*Graduate School of Science and Technology, Seoul National University, Republic of Korea
{hyejin210, park73}@snu.ac.kr
**Advanced Institutes for Convergence Technology, Republic of Korea
***Mechatronics Team, Institute for Advanced Engineering, Republic of Korea
ljm1918@iae.re.kr

***Abstract*— As more sophisticated robots are being developed, robots are increasingly expected to execute many types of task sets sequentially or simultaneously. When these multiple task sets are executed, the controllers of the robots should be able to deal dynamically with task changes. Especially during the task transitions, there will be discontinuous behaviors of robots in the absence of careful consideration of control. In this paper, a task transition approach is proposed to provide continuous task transitions and to ensure stable behavior of robots within the operational space control framework. In the proposed approach, the control law is not modified but the control command is composed using intermediate desired values. In this paper, a continuous task transition algorithm is applied for singularity avoidance and for joint limit avoidance purposes during the control of a 6-DOF manipulator to demonstrate its performance.**

***Keywords*— Task Transition, Operational Space Control, Singularity Avoidance, Joint Limit Avoidance**

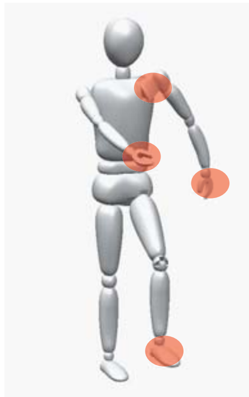


Fig. 1. Illustration of a humanoid robot capable of multiple tasks. The circles on the hands, foot, and shoulder are possible task points of the humanoid robot. A smooth task transition algorithm is important, as activation and deactivation of some of these tasks can occur sequentially or simultaneously depending on the situation.

1. Introduction

In recent years, the application of robotic systems has increased in various fields. Examples include service robots, medical robots, and robots that operate in hazardous environments, including traditional robots engaged in automation tasks. The robots in these new applications are being developed into more sophisticated forms to perform complex tasks. At the same time, these robots are expected to perform not only one specific task but also various tasks in sequence or simultaneously. Therefore, the management of multiple tasks [1], [2] and control of tasks with priorities [3], [4], [5] are currently significant issues. Other areas of study involve the simultaneous or consecutive execution of tasks, such as controlling various tasks and postures of a humanoid robot in a dynamic environment [6], [7] (Figure 1).

There are many cases involving the execution of multiple task sets in an interactive environment, such as various motion tasks [8], constraints [9], [10], joint limit avoidance [10], singularity avoidance [11], and visual servoing [12]. Stable and continuous behavior for dynamic transitions among these tasks becomes a significant issue when robots are controlled in an interactive environment [13].

The transitions among task sets, such as additions and removals, are considered in this paper. Transition approaches among task sets are developed by modifying the control schemes for these tasks [9], [10]. However, these approaches are limited in the case of task sets with different priorities. A transition algorithm with intermediate desired values [13] is proposed for kinematic control of robots to overcome the limitations. In this paper, a task transition algorithm for kinematic control in [13] is extended so that it can be used with the operational space control framework. The use of a task transition algorithm is of importance especially as it pertains to the operational space control framework because the framework does not use inverse kinematics or joint trajectories. Therefore, how a smooth transition among multiple tasks is implemented for the operational space control framework is an important issue.

The proposed algorithm in this paper applies the intermediate desired value approach [13] to the operational space control framework. First, the control inputs for task sets are redefined by using the intermediate desired values and this is extended to cases of prioritized task sets. The proposed approach enables transitions to be performed continuously without modification of the control framework. In this paper, a continuous task transition algorithm is applied to singularity avoidance and joint limit avoidance as examples to demonstrate its performance and effectiveness.

†Jaeheung Park is the corresponding author. park73@snu.ac.kr

2. Task Transition Approach

The equations of motion for a manipulator are described by

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

where q and Γ are the vector of the joint angles and torques, respectively. The terms $A(q)$, $b(q,\dot{q})$, and $g(q)$ denote the mass/inertia matrix, the Coriolis/centrifugal effects, and the gravity torques, respectively. To execute a task set x_1, the equations of motion in the operational space can be derived from Equation (1) [14]:

$$\Lambda_1(q)\ddot{x_1}+\mu_1(q,\dot{q})+p_1(q)=F_1, \tag{2}$$

where Λ_1, μ_1, and p_1 are the inertia/mass matrix, the Coriolis and centrifugal force, and the gravity force in the operational space of the task x_1, respectively. The control force F_1, can be determined using the operational space dynamics by

$$\begin{aligned} F_1 &= \Lambda_1(q)\tilde{f}_1 \\ &= \Lambda_1(q)\{f_1^*+\eta_1(q,\dot{q})+\zeta_1(q)\} \end{aligned} \tag{3}$$

and

$$\eta_1(q,\dot{q})=\Lambda_1^{-1}(q)\mu_1(q,\dot{q}), \quad \zeta_1(q)=\Lambda_1^{-1}(q)p_1(q), \tag{4}$$

where f_1^* is the control input for a unit-mass system or the desired acceleration for the task. The joint torques can then be computed by

$$\Gamma_1=J_1^T F_1. \tag{5}$$

where J_1 is the Jacobian corresponding to the task set x_1. The subscript 1 at Γ_1 denotes the task set x_1.

The joint torques may be discontinuous when a new task set x_2, is inserted. Once the new task is added, the command joint torques become

$$\Gamma \;=\; J^T F, \tag{6}$$

where

$$J=\begin{pmatrix} J_1 \\ J_2 \end{pmatrix}, \quad F=\begin{pmatrix} F_1 \\ F_2 \end{pmatrix}. \tag{7}$$

The change in the torque command from Γ_1 to Γ can lead to rapid discrete behavior of the task set x_2. That is, there may be discontinuous motion or unstable behavior during the transition to execute both task sets. This discrete change of the control input will also occur when removing or changing task sets.

To remove this discontinuity, intermediate desired values can be designed as a control input for x_2. That is, the task command can be designed such that the control torque would not change rapidly, as follows:

$$\Gamma=J^T\Lambda\tilde{f}^i, \quad \tilde{f}^i=\begin{pmatrix} \tilde{f}_1^i \\ \tilde{f}_2^i \end{pmatrix} \tag{8}$$

and

$$\begin{aligned} \tilde{f}_1^i &= \tilde{f}_1 \\ \tilde{f}_2^i &= h_2\tilde{f}_2+(1-h_2)J_2A^{-1}J_1^T\Lambda_1\tilde{f}_1 \end{aligned} \tag{9}$$

where h_2 is an activation parameter varying from 0.0 to 1.0 when inserting the task set x_2. The superscript i denotes the intermediate value for each parameter. The torque input is then computed by

$$\Gamma=J^T\Lambda\tilde{f}^i, \tag{10}$$

where

$$J(q)=\begin{pmatrix} J_1(q) \\ J_2(q) \end{pmatrix}. \tag{11}$$

This torque input Γ is continuous if the activation parameter changes from 0.0 to 1.0 continuously.

The proposed approach can be extended to other transition cases, i.e., when some of the tasks are removed. With the activation parameter h_1 defined for the first task set x_1, the task specifications can be described as

$$\begin{aligned} \tilde{f}_1^i &= h_1\tilde{f}_1+(1-h_1)J_1A^{-1}J_2^T h_2\Lambda_2\tilde{f}_2 \\ \tilde{f}_2^i &= h_2\tilde{f}_2+(1-h_2)J_2A^{-1}J_1^T h_1\Lambda_1\tilde{f}_1. \end{aligned} \tag{12}$$

Continuous transitions among non-prioritized task sets can be ensured by Equations (10) and (12) with continuous activation parameters h_1 and h_2, corresponding to each task set.

When there are three task sets with no priorities, the approach of the intermediate desired values for two task sets can be extended to

$$\begin{aligned} \tilde{f}_1^i &= h_1\tilde{f}_1+(1-h_1)J_1A^{-1}\Gamma_1^* \\ \tilde{f}_2^i &= h_2\tilde{f}_2+(1-h_2)J_2A^{-1}\Gamma_2^* \\ \tilde{f}_3^i &= h_3\tilde{f}_3+(1-h_3)J_3A^{-1}\Gamma_3^* \end{aligned} \tag{13}$$

where Γ_m^* denotes the torque input for task sets with intermediate desired values excluding the mth task set. The generalization to n tasks can be similarly derived.

3. Task Transition Approach for Prioritized Tasks

The proposed continuous transition approach can be applied to multiple task sets *with priorities*. The solution in the previous section can be rearranged to be applied to the prioritized task sets. For instance, we can consider two task sets with different priorities, x_1 and x_2, where a newly added task set x_1 has higher priority than the primal task set, x_2. The control input only for the primal task set x_2, is

$$\Gamma_2={J_2}^T\Lambda_2\tilde{f}_2. \tag{14}$$

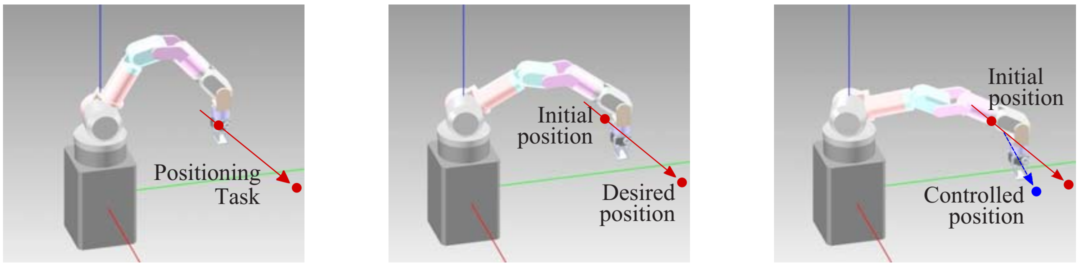
When the task set x_1 with higher priority is inserted, the control torque [6] is

$$\Gamma={J_1}^T\Lambda_1\tilde{f}_1+N_1^T\Gamma_0 \tag{15}$$

where

$$\begin{aligned} N_1^T &= (I-J_1^T\bar{J_1}^T) \\ \Gamma_0 &= J_2^T\Lambda_{2|1}\tilde{f}_2 \\ \Lambda_{2|1}^{-1} &= J_2N_1A^{-1}N_1^TJ_2^T. \end{aligned} \tag{16}$$

When the control torque is changed from Equation (14) to Equation (15), discontinuous behavior will occur during transition. Therefore, the proposed transition approach can



(a) (b) (c)

Fig. 2. Snapshots from the singularity avoidance experiment: (a) The position control of the end-effector was executed while the orientation task was also controlled as a task with higher priority. (b) The position of the end-effector followed a trajectory toward the desired position, which was out of the workspace. (c) Final configuration of the manipulator. The manipulator has moved to the desired position as much as possible while avoiding a singularity.

be applied to ensure continuity using intermediate desired values, as follows:

$$\Gamma = {J_1}^T \Lambda_1 \tilde{f}_1^i + N_1^T \Gamma_0 \tag{17}$$

where

$$\begin{aligned} \tilde{f}_1^i &= h_1 \tilde{f}_1 + (1-h_1) J_1 A^{-1} J_2^T \Lambda_2 \tilde{f}_2 \\ \Gamma_0 &= J_2^T \Lambda_{2|1} \tilde{f}_2. \end{aligned} \tag{18}$$

This formula, as expressed in Equations (17) and (18), provides a continuous transition among prioritized task sets when the task set x_1 is inserted at a higher priority. A smooth transition, the insertion of the task set x_1, can be realized by changing the activation parameter h_1 from 0.0 to 1.0 continuously.

The primal task x_2 can also have an activation parameter h_2 such that these two task sets can be added, removed or swapped by continuous activation parameters.

$$\Gamma = {J_1}^T \Lambda_1 \tilde{f}_1^i + (J_2 N_1)^T \Lambda_{2|1} \tilde{f}_2^i \tag{19}$$

where

$$\begin{aligned} \tilde{f}_1^i &= h_1 \tilde{f}_1 + (1-h_1) J_1 A^{-1} {J_2}^T h_2 \Lambda_2 \tilde{f}_2 \\ \tilde{f}_2^i &= h_2 \tilde{f}_2 + (1-h_2) J_2 A^{-1} {J_1}^T h_1 \Lambda_1 \tilde{f}_1. \end{aligned} \tag{20}$$

The intermediate desired value approach for two task sets can be extended to the case of three task sets. For three task sets,

$$\Gamma = \Gamma_1^i + \Gamma_2^i + \Gamma_3^i \tag{21}$$

where

$$\begin{aligned} \Gamma_1^i &= J_1^T \Lambda_1 \tilde{f}_1^i \\ \Gamma_2^i &= (J_2 N_1)^T \Lambda_{2|1} \tilde{f}_2^i \\ \Gamma_3^i &= (J_3 N_{2|1} N_1)^T \Lambda_{3|2|1} \tilde{f}_3^i, \end{aligned} \tag{22}$$

and

$$\begin{aligned} \tilde{f}_1^i &= h_1 \tilde{f}_1 + (1-h_1) J_1 A^{-1} (\Gamma^*_{\{1,prio\}}) \\ \tilde{f}_2^i &= h_2 \tilde{f}_2 + (1-h_2) J_2 A^{-1} (\Gamma^*_{\{2,prio\}}) \\ \tilde{f}_3^i &= h_3 \tilde{f}_3 + (1-h_3) J_3 A^{-1} (\Gamma^*_{\{3,prio\}}). \end{aligned} \tag{23}$$

The term $\Gamma^*_{\{m,prio\}}$ denotes the computed torque for all task sets, excluding the task set x_m with given priorities and activation parameters. At this point, it can be similarly extended to the case of n tasks.

4. Experimental Results

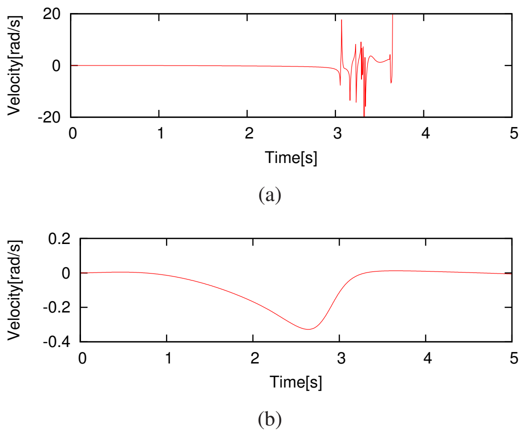
The proposed approach is verified using a 6-DOF manipulator called Roman-MD. The experiments were performed in a simulation environment called ROBOTICSLAB[15]. The experiment was conducted to prove the efficiency and robustness of the proposed approach in singularity avoidance and joint limit. In these experiments, the control input for position and orientation tasks is composed as follows:

$$f^* = k_p(x_d - x) - k_v \dot{x}, \tag{24}$$

where x, x_d, and $\dot{x}$ are the current state, desired value, and derivative of the current state of the task. The terms k_p and k_v are the positive gains of the PD control.

4.1 *Singularity Avoidance*

The proposed approach can be very effective in the application of singularity avoidance in the operational space control framework. The control while avoiding a singularity is implemented using the proposed continuous task transition algorithm by deactivating the task along the singular direction.



(a) (b)

Fig. 3. The sixth joint velocity near the singular configuration: (a) Oscillatory and unstable behavior due to a singularity without using the proposed approach. The simulation is terminated at around 3.7 seconds due to the high oscillation. (b) Smooth control using the proposed continuous transition algorithm near a singularity. The positioning task of the end-effector is smoothly executed while avoiding a singularity successfully.

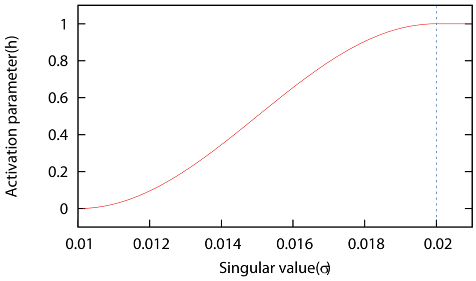


Fig. 4. The activation parameter of the task is designed as a function of the singular value by singular value decomposition (SVD).

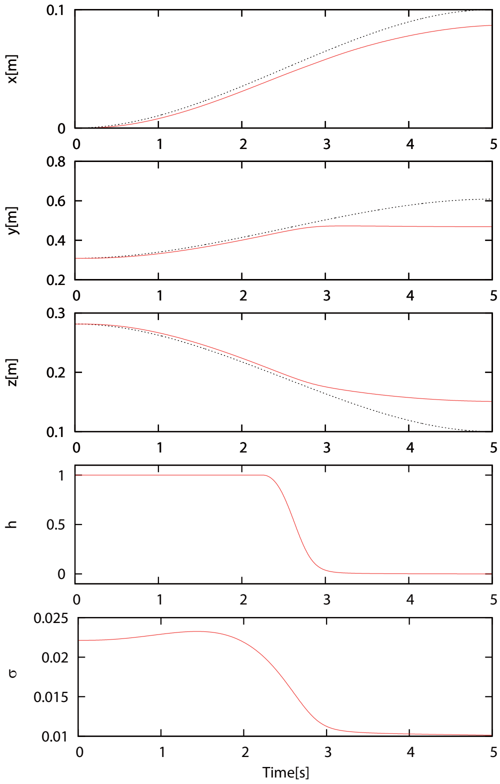


Fig. 5. Plots of the position, activation parameter, and singular value during the singularity avoidance experiment. It can be observed that the activation parameter starts to decrease at $\sigma = 0.02$. Once it is in the buffer region, the position cannot fully follow the desired trajectory due to the continuous deactivation of the positioning task in the singular direction.

In this experiment, the manipulator approaches the singularity when the end-effector is controlled to follow a trajectory which is beyond the workspace of the robot (Figure 2). During the positioning task of the end-effector, the orientation is controlled to be vertical as a task with higher priority. First, an experiment without using our transition algorithm is conducted to show the oscillatory unstable behavior near a singularity. Figure 3 shows plots of the velocity of the sixth joint near a singularity. As the positioning task is performed to reach the singular region, all joints start to oscillate.

This phenomenon is resolved by the proposed transition approach. In the application to a singularity, the transition is dependent on the singular values by singular value decomposition (SVD). An activation buffer is defined as the region of the singular value of the inverse of the inertia matrix in the operational space.

Using the hierarchical control structure shown in Equation (19), the orientation is assigned to be the task with the highest priority and the positioning task is second. In this experiment, the positioning task becomes the singular such the corresponding inertia matrix is considered for a singularity. The inverse of the inertia matrix of the positioning task can be decomposed into

$$J_p A^{-1} N_o^T J_p^T = U_p \Sigma_p U_p^T, \tag{25}$$

where J_p is the Jacobian for the position, J_o is the Jacobian for the orientation, the null-space projection matrix is $N_o^T = I - J_o^T \bar{J}_o^T$, Σ is a diagonal matrix with singular values, and U_p is the rotation matrix [16], [17]. When one of the singular values is less than a threshold, the corresponding direction is controlled in the buffer region. That is, the position control along that direction is continuously deactivated as the corresponding singular value decreases. For instance, if the last singular value is less than a threshold, U_p is decomposed into $[U_{p1}\ U_{p2}]$. The Jacobian for the singular direction is then defined as $U_{p2}^T J_p$.

When the singular value enters the buffer region, the activation parameter for the singular direction varies from 1.0 to 0.0 to deactivate the task along the singular direction, as plotted in Figure 4. The activation parameters for the positioning task of the end-effector except for the singular direction remain at 1.0 throughout the experiment because it is always fully activated. The activation parameter over the singular value in Figure 4 is designed to be continuous without any abrupt changes [13].

Figures 2 and 5 show snapshots and data plots from this experiment, respectively. The buffer of the singular values is set from 0.01 to 0.02 (Figure 4). It can be observed that the positioning task was smoothly deactivated along the singular direction, whereas the end-effector was still controlled to reach the desired position in other directions. Therefore, the positioning task is executed to the greatest extent possible while avoiding the singularity.

4.2 *Joint Limit Avoidance*

The task of joint limit avoidance is considered using the task transition algorithm in this section. When one of the joints moves close to its joint limit, the joint limit avoidance task is inserted. This should be considered as a task with a higher priority than the others.

In the experiment, the first joint enters the joint limit buffer region while the position and orientation of the end-effector are controlled. The upper and lower limits for the first joint are 0.75 (rad) and -0.75 (rad) in this example. The task of avoiding the joint limit is then added to the existing position and orientation task using the activation parameter h_{jl} over the buffer. Here, h_{jl} is defined as a function of the corresponding joint angle. When the first joint is in the activation buffer, the activation parameter is increased from 0.0 at the beginning of the buffer to 1.0 at the end of buffer. However, the activation parameter for the existing task of the end-effector, h_e, remains at 1.0 throughout the experiment because it is always fully activated.

As the new task of avoiding the joint limit is activated as a task with a higher priority, the positioning task is executed as much as the manipulator can achieve while avoiding the joint limit. Later, the end-effector of the manipulator moves to return to the initial position. The first joint is out of the joint limit buffer and the joint limit task is removed accordingly.

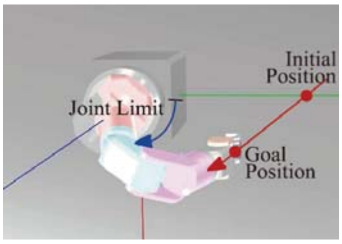


Fig. 6. Snapshots from the joint limit avoidance experiment. The joint limit avoidance task is considered to have a higher priority level.

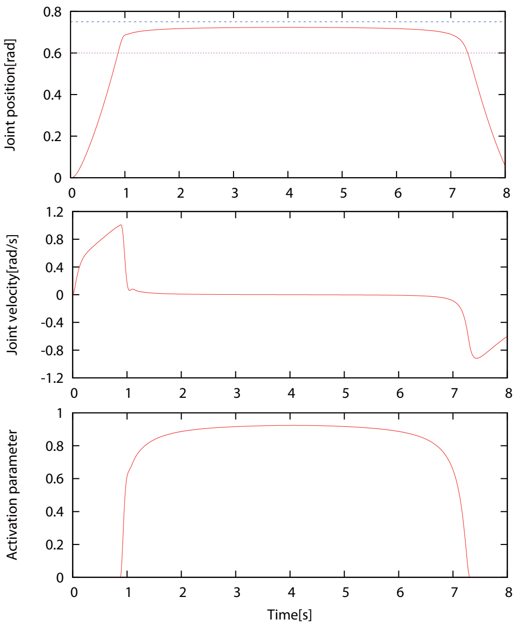


Fig. 7. Position and velocity of the first joint and activation parameter during the joint limit avoidance experiment. It can be observed that the activation parameter starts to increase at approximately 0.9*sec*.

The plots in Figure 7 show the stable and smooth transition of the position, velocity, and activation parameter of the corresponding joint limit task. It should be noted that the transition algorithm is effectively applied to both the insertion and the removal of the joint limit task.

5. Conclusion

A continuous transition approach among multiple tasks is proposed for the dynamic control of a robot. The intermediate desired value approach for kinematic control is successfully applied for the operational space control framework by composing the command inputs for the multiple tasks. Also, experiments on singularity avoidance and joint limit avoidance were conducted to demonstrate the effectiveness of the proposed approach in the operational space control framework.

The occurrence of discontinuous behavior due to an abrupt change of the command torque at the task transition is resolved by the proposed approach. The approach can also deal with a hierarchical task set by activating or deactivating task sets continuously. Therefore, transitions such as the insertion or removal of tasks can be performed with various priority levels. The experimental results demonstrate the performance of the proposed approach. Having successfully implemented the approach for motion control in the operational space control framework, we are currently investigating further developments of this approach to include contact force control.

References

[1] B. Sicilliano and J.-J. E. Slotine, “A general framework for managing multiple tasks in highly redundant robotic systems,” in *IEEE Int. Conf. on Advanced Robotics (ICAR’91)*, Napoli, Italy, 1991, pp. 1211–1216.

[2] N. Mansard and F. Chaumette, “Task sequencing for high-level sensor-based control,” *IEEE Trans. Robot. Automat.*, vol. 23, pp. 60–72, Feb. 2007.

[3] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, “Task-priority based redundancy control of robot manipulators,” *Int. Journal of Robotics Research*, vol. 6, pp. 3–15, June 1987.

[4] P. Baerlocher and R. Boulic, “Task-priority formulations for the kinematic control of highly redundant articulated structures,” in *IEEE Int. Conf. on Intelligent Robots and Systems (IROS’98)*, Victoria, B.C., Canada, Oct. 1998, pp. 323–329.

[5] D. N. Nenchev and Z. M. Sotirov, “Dynamic task-priority allocation for kinematically redundant robotic mechanisms,” in *IEEE Int. Conf. on Intelligent Robots and Systems (IROS’94)*, Munich, German, Sept. 1994, pp. 518–524.

[6] 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, pp. 29–43, 2004.

[7] E. Yoshida, I. Belousov, C. Esteves, and J.-P. Laumond, “Humanoid motion planning for dynamic tasks,” in *IEEE Int. Conf. on Humanoid Robots*, Tsukuba, Japan, Dec. 2005, pp. 1–6.

[8] N. Mansard, A. Remazeilles, and F. Chaumette, “Continuity of varying-feature-set control laws,” *IEEE Trans. on Automatic Control*, vol. 54, pp. 2493–2505, Nov. 2009.

[9] N. Mansard and O. Khatib, “Continuous control law from unilateral constraints,” in *IEEE Int. Conf. on Robotics and Automation (ICRA’08)*, Pasadena, CA, USA, May 2008, pp. 3359–3364.

[10] N. Mansard, O. Khatib, and A. Kheddar, “A unified approach to integrate unilateral constraints in the stack of tasks,” *IEEE Trans. on Robotics*, vol. 25, pp. 670–685, June 2009.

[11] F. Chaumette, P. Rives, and B. Espiau, “Positioning of a robot with respect to an object, tracking it and estimating its velocity by visual servoing,” in *IEEE Int. Conf. on Robotics and Automation (ICRA’91)*, Sacramento, CA, USA, Apr. 1991, pp. 2248–2253.

[12] S. Chiaverini, “Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators,” *IEEE Trans. on Robotics*, vol. 13, pp. 398–410, June 1997.

[13] J. Lee, N. Mansard, and J. Park, “Intermediate desired value approach for continuous transition among multiple tasks of robots,” in *IEEE Int. Conf. on Robotics and Automation (ICRA’11)*, Shanghai, China, May 2011, pp. 1276–1282.

[14] O. Khatib, “A unified approach for motion and force control of robot manipulators: The operational space formulation,” *IEEE Trans. Robot. Automat.*, vol. 3, pp. 43–53, Feb. 1987.

[15] SimLab. Roboticslab. [Online]. Available: http://www.rlab.co.kr/

[16] G. Strang, *Linear Algebra and Its Applications, Fourth Edition*. 10 Davis Drive, Belmont, CA 94002-3098, USA: Thomson Brooks/Cole, 2006.

[17] J. Park, “Control strategies for robots in contact,” Ph.D. dissertation, Stanford University, Stanford, CA, USA, Mar. 2006. [Online]. Available: http://ai.stanford.edu/ park73/papers/Jaeheung-thesis.pdf