

Intermediate desired value approach for continuous transition among multiple tasks of robots

Jaemin Lee, Nicolas Mansard, Jaeheung Park

▶ **To cite this version:**

Jaemin Lee, Nicolas Mansard, Jaeheung Park. Intermediate desired value approach for continuous transition among multiple tasks of robots. IEEE International Conference on Robotics and Automation (IEEE ICRA 2011), May 2011, shanghai, China. 7p., 10.1109/ICRA.2011.5980194 . hal-00667212

HAL Id: hal-00667212
https://hal.science/hal-00667212

Submitted on 7 Feb 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Intermediate Desired Value Approach for Continuous Transition among Multiple Tasks of Robots

Jaemin Lee*, Nicolas Mansard**, and Jaeheung Park*†
*Department of Intelligent Convergence Systems, Seoul National University, Republic of Korea.
{ljm1918, park73}@snu.ac.kr
**Laboratoire d'Architecture et d' Analyse des Systems, Universite de Toulouse, France.
nicolas.mansard@laas.fr

***Abstract*— As the capability of robots is getting improved, more various tasks are expected to be performed by the robots. Complex operation of the robots can be composed of many different tasks. These tasks are executed sequentially, simultaneously, or in a combined way of both. This paper discusses the transition issue among multiple tasks on how the transition can be effectively and smoothly achieved. The proposed approach is to compose intermediate desired values to smooth the transitions rather than to modify control laws. The approach can be practically used on robotic systems without modification on their specific control algorithms. In this paper, multi-points control and joint limit avoidance are performed as applications of the proposed approach.**

I. Introduction

The robots are mostly programmed to execute a specific pre-defined set of tasks in a well defined environment. However, as more complex robots are designed to operate in dynamic environments, various tasks or set of tasks should be executed sequentially or simultaneously [1], [2]. Dealing with these situations, it is important to design strategies for transition among various tasks [3], [4], [5].

Considering general task transitions, this paper is focused on the situation when a task is added or deleted. Once established, this approach can be used for many task transition situations. Such cases can occur also in traditional applications such as dealing with joint limit where joint limit avoidance can be defined as a task in specific regions [6]. Feature tracking in visual servoing is another application where some of the features are lost or added during operation [6], [7], [8]. Also, Singularity avoidance in manipulation [9] and switching behaviors of humanoid robots would be also one of the cases [11], [12], [13].

Without proper consideration of transition, there will be discrete and abrupt changes at the input of the robot, which is typically either desired joint velocity or joint torque. The goal of the paper is to provide an intelligent transition method that enables the robot to perform smooth task transition.

In this paper, the concept and validation are presented in a kinematic control approach, where joint velocities are the inputs to the robot given desired task space velocities. In the following section, the source of discontinuity is explained. One approach dealing with this discontinuity is then proposed by introducing intermediate desired values as a new task specification.

The proposed approach is compared with the continuous inverse solution in [14], which modifies the inversion process. The continuous inverse approach is to compose a control law by modifying the inverse operation with an activation matrix. It is revealed that the two approaches are equivalent in the case of equal priority tasks. However, the new approach can be directly applied to hierarchically constructed tasks, where the continuous inverse solution is limited. Also, the new approach is easily applicable to many existing platforms since it does not require to change control or inverse process.

In addition, it is briefly discussed the case when there is not enough DOF for two tasks with different priorities. In such case the lower-priority task cannot be fully performed. Its control result is discussed to provide an intuition of the priority-based control and the effectiveness of the proposed approach for prioritized multiple tasks.

The experimental validations of the proposed approach were conducted in two scenarios: multi-point control using a 3 DOF planar robot in simulation and joint limit avoidance using a 7 DOF physical robot. The smooth and effective task transitions are demonstrated in both cases.

II. Discontinuity from Task Addition or Removal

When there are two tasks or features, x_1 and x_2, the corresponding Jacobians are defined as J_1 and J_2.

$$\dot{x}_1 = J_1\dot{q}, \quad \dot{x}_2 = J_2\dot{q} \tag{1}$$

When only one task, x_1, is to be controlled, the associated joint velocity, $\dot{q}_1$, can be computed as

$$\dot{q}_1 = J_1^+\dot{x}_1 \tag{2}$$

where $(.)^+$ denotes a pseudo-inverse of the quantity. The inverse could be replaced by any of the generalized inverses.

Now, if another task, x_2, is to be controlled together with x_1, the joint velocity can be computed as

$$\dot{q} = J^+\dot{x} \tag{3}$$

where

$$J = \begin{pmatrix} J_1 \\ J_2 \end{pmatrix}, \quad \dot{x} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix}. \tag{4}$$

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

The joint velocity for both tasks (Equation (3)) might be *discontinuous* from the joint velocity for only the first task (Equation (2)) at the instance when the second feature is inserted.

One explanation for this discontinuous behavior is that it comes from the property of pseudo-inverse. The pseudo-inverse computes a solution to meet the feature specification and minimize the 2-norm of $\dot{q}$. Obviously, when we achieve one feature or two features the solution would not be continuous.

More physically meaningful explanation is as follows. When we control the first task or feature, the computed joint velocity, q_1, indirectly, affect the task, x_2, although this task is not interested during the execution. Then, the problem is that we later attempt to control or specify this feature with different values all of sudden. Therefore the joint velocity will show a discrete change.

Fundamentally, the problem is that the solution, $\dot{q}_1$, from Equation (2) generates a value for the second task as $J_2\dot{q}_1$ since we did not specify it. Then, when we specify a desired value, $\dot{x}_2$,

$$\dot{x}_2 \neq J_2\dot{q}_1 \tag{5}$$

Therefore, when we add this task, x_2, the specification for x_2 in Equation (4) should be continuous from the value, $J_2\dot{q}_1$. Otherwise, the pseudo-inverse will give a solution with a discrete change. The discrete change would occur during task removal due to the same reason.

III. Intermediate Desired Value Approach

A. Concept of Continuous Transition Strategy

From the observation in the previous section, a proposed approach is to design a task specification of $\dot{x}_2$ smoothly from $J_2\dot{q}_1$, where $\dot{q}_1$ is from Equation (2), such that the solution would not show any abrupt change.

$$\dot{x}^i = \begin{pmatrix} \dot{x}_1^i \\ \dot{x}_2^i \end{pmatrix} \tag{6}$$

and

$$\begin{aligned} \dot{x}_1^i &= \dot{x}_1 \\ \dot{x}_2^i &= h_2\dot{x}_2 + (1-h_2)J_2J_1^+\dot{x}_1 \end{aligned} \tag{7}$$

where h_2 is an activation variable from 0 to 1 when the task, x_2, is inserted. The superscript, i, denotes the intermediate value of the quantity. Then the joint velocity is

$$\dot{q} = J^+\dot{x}^i. \tag{8}$$

This solution is continuous if we change h_2 from 0 to 1 continuously.

B. Generalization

An activation parameter for the first task, x_1, can also defined as h_1. Then,

$$\dot{q} = J^+\dot{x}^i \tag{9}$$

$$\begin{aligned} \dot{x}_1^i &= h_1\dot{x}_1 + (1-h_1)J_1J_2^+h_2\dot{x}_2 \\ \dot{x}_2^i &= h_2\dot{x}_2 + (1-h_2)J_2J_1^+h_1\dot{x}_1. \end{aligned} \tag{10}$$

This solution is in fact the same as the one for one task with activation matrix, H, in [16]. Refer to Appendix I for the proof of equivalence.

C. More than Two tasks

When there are tasks with more than two activation parameters, Equation (10) can be extended. For example, the solution for the tasks with three activation parameters is the following:

$$\dot{q} = J^+\dot{x}^i \tag{11}$$

And

$$\begin{aligned} \dot{x}_1^i &= h_1\dot{x}_1 + (1-h_1)J_1\dot{q}_{\{h_2,h_3\}} \\ \dot{x}_2^i &= h_2\dot{x}_2 + (1-h_2)J_2\dot{q}_{\{h_1,h_3\}} \\ \dot{x}_3^i &= h_3\dot{x}_3 + (1-h_3)J_3\dot{q}_{\{h_1,h_2\}}, \end{aligned} \tag{12}$$

where $\dot{q}_{\{h_m,h_n\}}$ denotes the solution for the tasks m and n with activation parameters of h_m and h_n. The derivation of Equations (11) and (12) is straight-forward from Equations (9) and (10).

IV. Intermediate Desired Value Approach for Prioritized Tasks

A new task may be added with a different priority from the existing tasks. The solution of the previous section can be re-arranged in this case. For example, when we had a task set as x_2 and the associated Jacobian, J_2, another higher prioritized task set, x_1, can be inserted, such as joint limits or collision avoidance. The controller for x_2 only was

$$\dot{q}_2 = J_2^+\dot{x}_2. \tag{13}$$

Now with the higher prioritized task, x_1,

$$\dot{q} = J_1^+\dot{x}_1 + N_1\dot{q}_0 \tag{14}$$

where

$$N_1 = I - J_1^+J_1 \tag{15}$$

$$\dot{q}_0 = (J_2N_1)^+(\dot{x}_2 - J_2J_1^+\dot{x}_1). \tag{16}$$

However, transition from Equation (13) to Equation (14) will have discontinuity. Therefore, a similar approach to the one introduced in the previous section can be applied.

$$\dot{q} = J_1^+\dot{x}_1^i + N_1\dot{q}_0^i \tag{17}$$

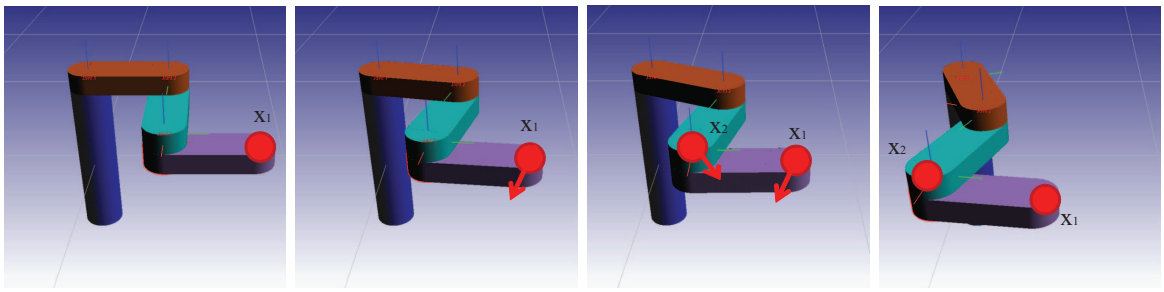
where

$$\begin{aligned} \dot{x}_1^i &= h_1\dot{x}_1 + (1-h_1)J_1\dot{q}_2 \\ \dot{q}_2 &= J_2^+\dot{x}_2 \\ \dot{q}_0^i &= (J_2N_1)^+(\dot{x}_2 - J_2J_1^+\dot{x}_1^i). \end{aligned} \tag{18}$$

The above approach provides a continuous solution for $0 \leq h_1 \leq 1$ because $\dot{q}_2$ from Equation (13) is the same as $\dot{q}$ from Equation (17) when $h_1 = 0$ (See Appendix II).

However, the above solution does not have an activation parameter for x_2. This could be implemented by introducing h_2.

$$\dot{q} = J_1^+\dot{x}_1^i + N_1\dot{q}_0^i \tag{19}$$



(a) (b) (c) (c)

Fig. 1. 3 DOF planar robot simulation: there are two task sets, x_1 and x_2. (a) initial position of 3-DOF planar robot (b) positioning task of the end effector with high priority, x_1, is being executed without the other task, x_2. (c) Positioning task of 2nd link, x_2, is added with a lower priority. (d) x_1 and x_2 are executed with priorities.

where

$$\begin{aligned}\dot{x}_1^i &= h_1\dot{x}_1 + (1-h_1)J_1\dot{q}_2^i \\ \dot{q}_2^i &= J_2^+ h_2\dot{x}_2 \\ \dot{q}_0^i &= (J_2N_1)^+(h_2\dot{x}_2 + (1-h_2)J_2J_1^+h_1\dot{x}_1 - J_2J_1^+\dot{x}_1^i).\end{aligned} \tag{20}$$

The equations (19) and (20) can be written as

$$\dot{q} = J_1^+\dot{x}_1^i + N_1(J_2N_1)^+(\dot{x}_2^i - J_2J_1^+\dot{x}_1^i) \tag{21}$$

where

$$\begin{aligned}\dot{x}_1^i &= h_1\dot{x}_1 + (1-h_1)J_1J_2^+h_2\dot{x}_2 \\ \dot{x}_2^i &= h_2\dot{x}_2 + (1-h_2)J_2J_1^+h_1\dot{x}_1\end{aligned} \tag{22}$$

A. What is the difference from the two approaches with or without priorities?

As we can see from Equation (21) and (22), the only difference in formulations for the approaches with or without priorities is the computational structure. The intermediate goal values, $\dot{x}^i$, are computed in the same procedure.

If the two tasks are independent, the two approaches with or without priorities should give the same result. The difference comes only if they are conflicting.

B. More than Two Tasks

The same extension approach as the previous solution applies. For the three task sets,

$$\dot{q} = \dot{q}_1^i + \dot{q}_2^i + \dot{q}_3^i \tag{23}$$

$$\begin{aligned}\dot{q}_1^i &= J_1^+\dot{x}_1^i \\ \dot{q}_2^i &= N_1(J_2N_1)^+(\dot{x}_2^i - J_2\dot{q}_1^i) \\ \dot{q}_3^i &= N_1N_{2|1}(J_3N_1N_{2|1})^+(\dot{x}_3^i - J_3(\dot{q}_1^i + \dot{q}_2^i))\end{aligned} \tag{24}$$

and

$$N_{2|1} = I - (J_2N_1)^+J_2N_1 \tag{25}$$

$$\begin{aligned}\dot{x}_1^i &= h_1\dot{x}_1 + (1-h_1)J_1\dot{q}_{\{h_2,h_3,prio\}} \\ \dot{x}_2^i &= h_2\dot{x}_2 + (1-h_2)J_2\dot{q}_{\{h_1,h_3,prio\}} \\ \dot{x}_3^i &= h_3\dot{x}_3 + (1-h_3)J_3\dot{q}_{\{h_1,h_2,prio\}},\end{aligned} \tag{26}$$

where $\dot{q}_{\{h_m,h_n,prio\}}$ denotes the solution for the tasks m and n with activation parameters of h_m, h_n, and given priorities.

V. Simulation and Experimental Results

The proposed approach for task transition has been verified in ROBOTICSLAB [17] simulation environment and physical robot. ROBOTICSLAB provides not only physics-based simulation environment but also real time control module and programmable interface for the user.

In the following sub-sections, the multiple point control with different priorities is demonstrated in simulation with a 3-DOF planar robot. Then, the end-effector control with joint limit avoidance is executed on a 7-DOF manipulator. During the experiments, the kinematic control law is implemented as

$$\dot{x} = -\lambda(x - x_d) \tag{27}$$

where x and x_d are the current and desired task, and λ is a positive gain for decreasing task error.

A. Task transition in a 3 DOF planar robot: Simulation

In this section, the proposed task transition approach is demonstrated in a 3 DOF planar robot when a new task inserted with a lower priority than that of an existing task. The purpose of the section is to demonstrate the important concepts in a relatively simpler case. Also, the effect of the priorities is explained and shown when the tasks are partially conflicting due to the lack of degrees of freedom of the robot.

The two positioning tasks are defined as positioning of the end-effector and the 2nd link (Figure 1).

$$\dot{x}_1 = \begin{pmatrix} \dot{x}_{e,x} \\ \dot{x}_{e,y} \end{pmatrix} \quad \dot{x}_2 = \begin{pmatrix} \dot{x}_{link2,x} \\ \dot{x}_{link2,y} \end{pmatrix} \tag{28}$$

where $(.)_{e,(.)}$ and $(.)_{link2,(.)}$ denote the tasks related to the end-effector and the end point of the 2nd link, respectively. The robot begins to control only the end-effector position, x_1. Later the 2nd link position control is inserted after 2 seconds (t = 4 sec) (Figure 2 and 3).

In Figure 2, the control results are plotted when the second task (positioning task of the 2nd link) is abruptly inserted as a low priority task. The results of the proposed continuous approach are plotted in Figure 3. The activation parameter h_2 increased from 0.0 to 1.0 for 0.5 seconds. The plot of

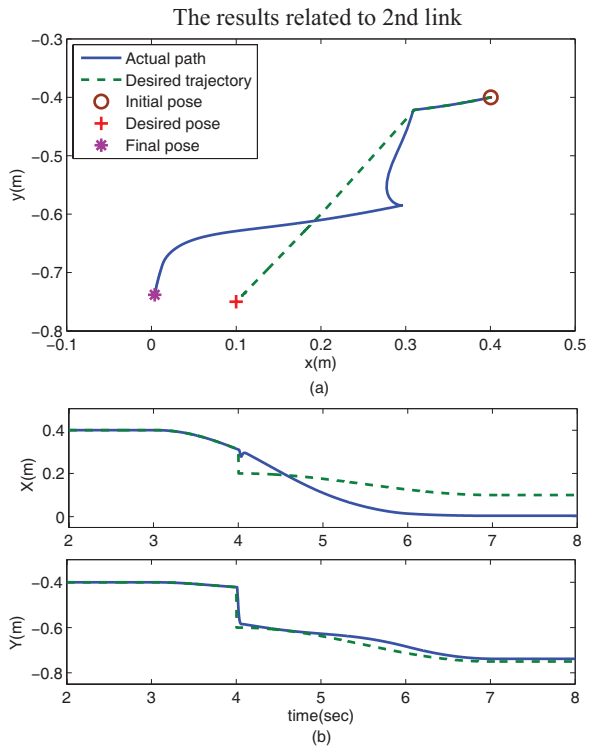


Fig. 2. Example of discontinuity of the control law when a task is inserted abruptly for the 3 DOF planar robot. (a) Position of 2nd link of 3 DOF planar robot during the control. (b) The inserted task set, x_2, is plotted over time. The desired position of 2nd link is out of reach due to the high prioritized task, x_1. The final position is the closest reachable position, as high prioritized task, x_1, is executed. As soon as x_2 is inserted, undesirable jerky motion occurs at the end-effector and 2nd link paths.

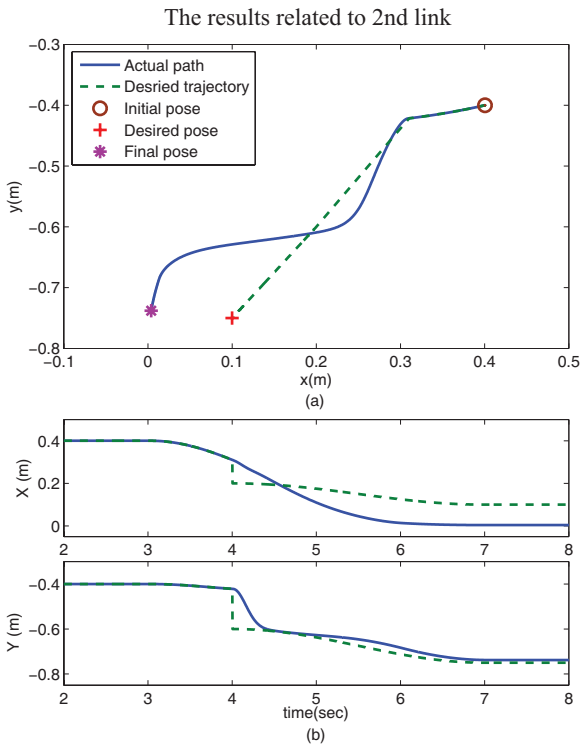


Fig. 3. Example of task transition executed by proposed control approach for the 3 DOF planar robot. (a) Position of the 2nd link of 3 DOF planar robot during the control. (b) The inserted task set, x_2, is plotted over time. The parameters of the simulation, initial position, desired position, trajectories, operating time and etc., are the same as the case of discontinuity (Figure 2). Compared to the result of the abrupt transition, the paths of the end-effector and 2nd link are smooth and there is no undesirable motion.

the first task is omitted for both cases since their tracking performances are good and almost the same.

The abrupt change in Figure 2 is successfully removed by the proposed continuous approach in Figure 3.

Note the fact that the inserted task is not executed perfectly due to the lack of degrees of freedom for both tasks. This phenomenon is explained from the global and instantaneous point of view in Figure 4 and 5, respectively. At the end of the control in Figure 4, the position of the 2nd link comes to the closest point to the desired position within the workspace, which is a circle centered at the end-effector position.

In the instantaneous view, the second task execution can be decomposed into the effect of higher-priority task and its control in the task null space (Figure 5).

$$\begin{aligned} \dot{x}_2 &= J_2 \dot{q} \\ &= \dot{x}_{2,prio} + \dot{x}_{2,null} \end{aligned} \tag{29}$$

where

$$\begin{aligned} \dot{x}_{prio} &= J_2 J_1^{+} \dot{x}_1 \\ \dot{x}_{null} &= J_2 N_1 \dot{q}_0. \end{aligned} \tag{30}$$

The term, $\dot{x}_{2,prio}$, is the effect from the control of the previous task with a higher priority. Then, the control in the null space, $\dot{x}_{2,null}$, set $\dot{x}_2$ to be closest to its desired value.

Although the inserted task cannot be fully performed because of not enough degrees of freedom for both tasks, it is demonstrated that the task is executed optimally under priority-based control. Using recursive control structure with the proposed approach, many tasks can be controlled partially or completely with various levels of priorities.

B. 7-DOF manipulator with elbow joint limit: Experiment

Joint limit task for a 7 DOF manipulator is controlled through the proposed approach in this section. It must be considered as a higher prioritized task than other given tasks. When the joint limit task is inserted, continuity of task transition can be ensured by the proposed approach.

1) Definition of Prioritized Tasks: In this section, two task sets are considered with priorities. One is a joint limit task and the other is consisted of position and orientation of end-effector. The priority of joint limit task is higher than that of task related to the end-effector.

$$\dot{x}_1 = \begin{pmatrix} \dot{q}_{limit} \end{pmatrix}, \quad \dot{x}_2 = \begin{pmatrix} v_e \\ \omega_e \end{pmatrix}, \tag{31}$$

where v_e and ω_e are the linear and angular velocities of the end-effector.

When the joint position is not in joint limit region, the end-effector positioning task is controlled by

$$\dot{q} = J_2^{+} \dot{x}_2. \tag{32}$$

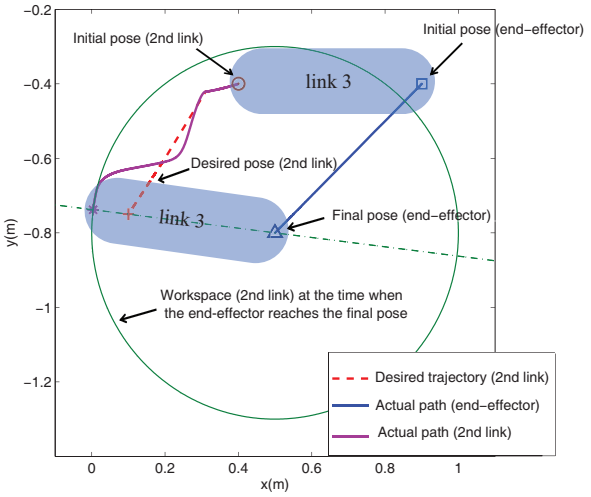


Fig. 4. Task execution of the two tasks: the end-effector, x_1, and the 2nd link, x_2, of the 3 DOF planar robot with different priorities. The final position of the 2nd link, which has the lower priority, does not reach its goal position but to the closest position from the desired position in the reachable positions (the circle centered at the final position of the end-effector).

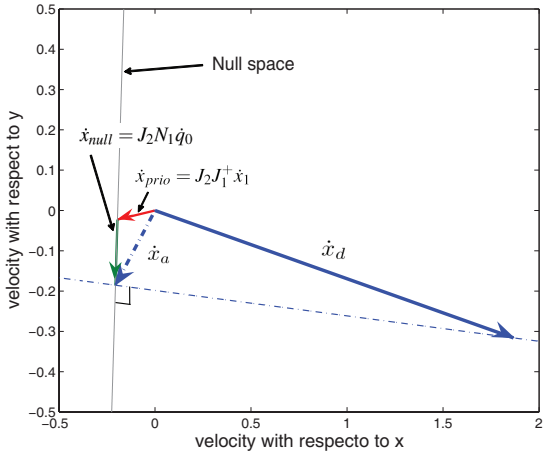


Fig. 5. Instantaneous task execution of the low-priority task, x_2, in the 3 DOF planar robot. The execution of the inserted task with the lower priority is consisted of the effect from the high-priority task and its control in the task null space. However, when there is not enough DOF for both tasks, the control of the low-priority task in the null space partially achieve its goal. The term $\dot{x}_d$ is the desired value for $\dot{x}_2$ and $\dot{x}_a$ is the actual command for $\dot{x}_2$ due to the lack of DOF and dependency between task sets.

Now when the joint limit task with high priority is inserted without intermediate desired value method, the kinematic control with the priorities provides the solution of

$$\dot{q} = J_1^+ \dot{x}_1 + N_1 (J_2 N_1)^+ (\dot{x}_2 - J_2 J_1^+ \dot{x}_1). \tag{33}$$

The result of this kinematic control through an abrupt change produces oscillatory behavior. The oscillation comes from rapid task transition at the activation border (Figure 8). It means that pushing toward and backward at the joint limit are iterated in sequence.

In the experiments, the end-effector of the robot is controlled to follow a line to reach a desired goal position. While the end-effector task is being executed, the Elbow joint reaches a joint limit buffer at right before 5 seconds (Figure 7, 8, and 9). Figure 8 shows instability due to the iterative abrupt changes in control due to the conflicting joint limit and end-effector control. The results of the proposed approach are plotted in Figure 9, which is explained in detail in the following sub-sections.

2) Activation Buffer: For the joint limit task execution, the activation parameter is defined as a function of the corresponding joint angle. In the activation buffer, the activation parameter increases from 0.0 at the beginning of the buffer to 1.0 at the limit. For inserting high-priority task, the activation parameter, h_1, is defined as

$$h_1 = \begin{cases} 1.0 & \text{if } q_{elbow} \geq q_{upperLimit} \\ f(q_{elbow}) & \text{if } q_{upperLimit} - \beta < q_{elbow} < q_{upperLimit} \\ 0.0 & \text{if } q_{lowerLimit} + \beta \leq q_{elbow} \leq q_{upperLimit} - \beta \\ g(q_{elbow}) & \text{if } q_{lowerLimit} < q_{elbow} < q_{lowerLimit} + \beta \\ 1.0 & \text{if } q_{elbow} \leq q_{lowerLimit} \end{cases} \tag{34}$$

where β is the buffer length ($\beta = 0.2(rad)$), $q_{upperLimit} = 1.6(rad)$, and $q_{lowerLimit} = -1.6(rad)$ in this example. The

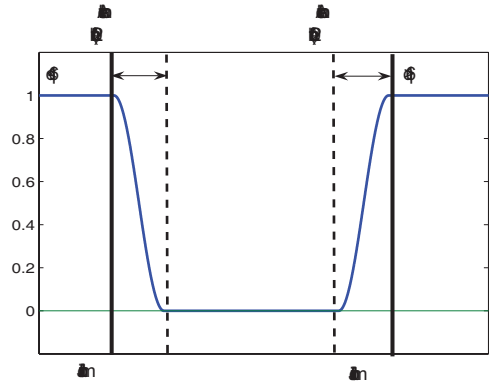


Fig. 6. Activation parameter of task transition, h_1, for the joint limit task in the 7 DOF manipulator. It is parameterized with the joint angle in the buffer zone.

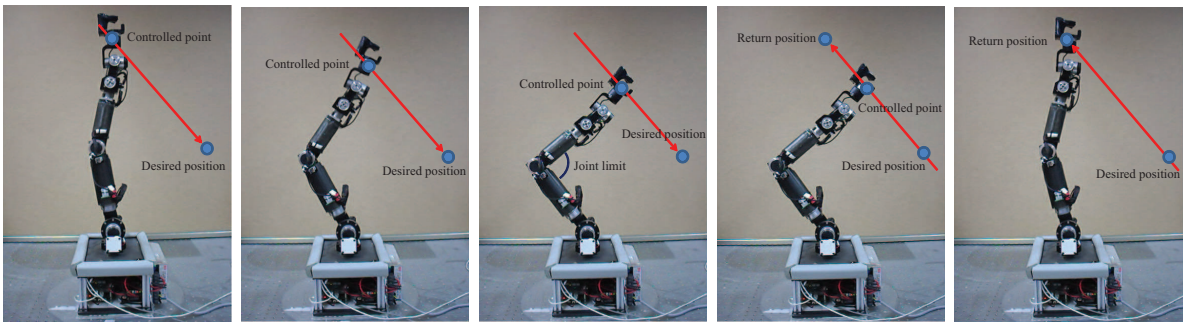
activation parameter is plotted in Figure 6. Equation (35) are activation functions varying in activation buffers.

$$f(q_{elbow}) = 0.5 + 0.5 sin(\frac{\pi}{\beta}(q_{elbow} - (q_{upperLimit} - \beta)) - \frac{\pi}{2})$$
$$g(q_{elbow}) = 0.5 + 0.5 sin(\frac{\pi}{\beta}(q_{elbow} - q_{lowerLimit}) + \frac{\pi}{2}) \tag{35}$$

3) Intermediate Desired Value: Discontinuity and instability of task execution occur when tasks are transitioned abruptly. For continuous task transition, the intermediate desired value for the joint limit task for the elbow is set to be

$$\dot{x}_1^i = h_1 \dot{x}_1 + (1 - h_1) J_1 J_2^+ \dot{x}_2 \tag{36}$$

where h_1 is the activation function defined in Equation (34). When the elbow joint is in the activation buffer zone, the joint



(a) (b) (c) (d) (e)

Fig. 7. Joint limit task execution of the 7 DOF manipulator with positioning and orientation tasks. (a) Initial position of 7-DOF manipulator (b) Positioning and orientation task, x_2, execution without joint limit (c) x_2 execution with joint limit (d) Return positioning task, x_2, execution as elbow joint is extricated from joint limit (e) Final position of 7-DOF manipulator

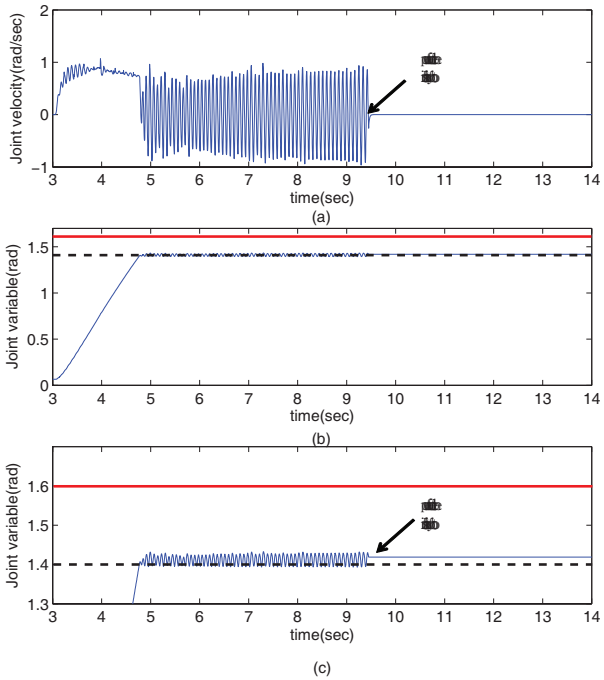


Fig. 8. Oscillation and instability due to the discrete task transition. When the activation border is reached, joint limit task is inserted with a higher priority, the oscillation starts due to the discrete task transition. The experiment is terminated at around 9.4 seconds due to the instability of the robot. (a) Joint velocity. (b) Elbow joint angle near the activation border. (c) Elbow joint angle plotted in a smaller scale (zoom-in view). The joint angle oscillates around activation border.

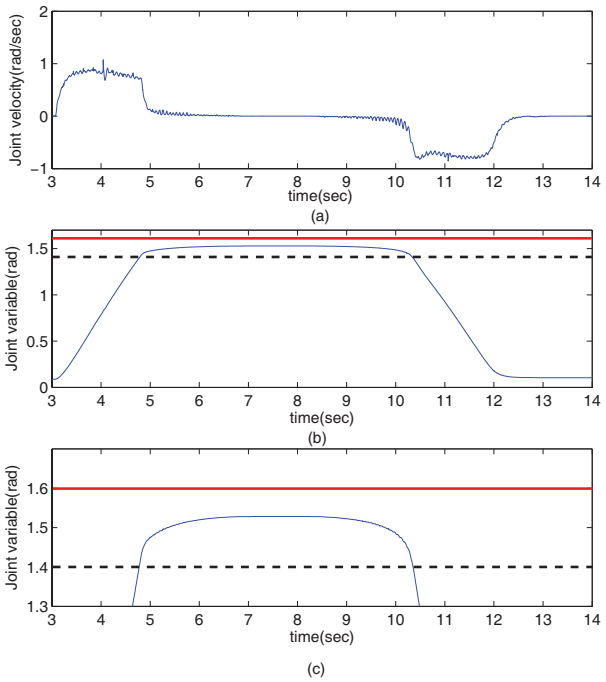


Fig. 9. Execution of joint limit task by the proposed continuous control law. The joint limit task is being executed successfully without any instability with the end-effector task simultaneously. (a) Joint velocity (b) Elbow joint angle: The joint angle converges within activation buffer.(c) Elbow joint angle plotted in a smaller scale (zoom-in view)

limit task has to be inserted without discontinuity and instability. In Figure 9, there are no discontinuity and instability in the case which is controlled by the proposed approach. The joint position of elbow has been successfully controlled at the border of activation buffer and the activation parameter has varied continuously with respect to the position of the elbow joint. In this case, the elbow joint does not reach its limit. In the cases when more joints enter into the joint limit regions, the same approach for more tasks can be simply used.

VI. Conclusion

The intermediate desired value approach for task transition is presented in this paper. The proposed approach is to provide continuity during transition using existing control frameworks with minimal performance compromise. The approach without priorities are shown to be equivalent to continuous inverse method [14]. Then, it is further extended to the cases of tasks with priorities.

When task transition is executed by the proposed approach, the transitioned tasks can be executed smoothly with various levels of priorities including high priority and low priority. When there is not enough DOF of the manipulator,

the task with a lower priority may not be fully performed but partially. The simulation result of such a case is shown to demonstrate that the proposed approach can be used successfully with the priority-based control framework.

We are currently working on extending the proposed approach for the dynamic control framework. And the issue of computation cost of the proposed approach will be investigated in the future.

VII. ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (No.2010-0005799) and Seoul R&BD Program (No.JP100106).

REFERENCES

[1] Y.Nakamura, H.Hanafusa and T.Yoshikawa, "Task-priority based redundancy control of robot manipulators", The Int. J. Robot. Res., Vol.6, No.2, 1987, pp. 3-15

[2] P.Baerlocher and R. Boulic, "Task-Priority Formulations for the Kinematic Control of Highly Redundant Articulated Structures", In Proc. of the Int. Conf. on Intelligent Robots and Systems, 1998, pp. 323-329

[3] Dragonir N.Nenchev and Zlatko M.Sotirov, "Dynamic Task-priority allocation for kinematically redundant robotic mechanisms". In Proc. of the Int. Conf. on Intelligent Robots and Systems, 1994, pp. 518-524

[4] B.Baerlocher and R.Boulic, "An Inverse Kinematic Architecture Enforcing an Arbitrary Number of strict priority levels", The Visual Computer, Vol.6, No.30, 2004, pp.402-417

[5] B.Siciliano and Jean-Jacques E.Slotine, "A General Framework for Managing Multiple Tasks in Highly Redundant Robotic Systems", In Proc. of the Int. Conf. on Advanced Robots, 1991, pp. 1211-1216

[6] 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, No.3, 2009, pp.670-685

[7] N.Mansard and F.Chaumette, "Task sequencing for sensor-based control", IEEE Trans. on Robotics, Vol.23, No.1, 2007, pp.60-72

[8] N.Mansard, A.Remazeilles and F.Chaumette, "Continuity of Varying-Feature-Set Control laws", IEEE Trans. on Automatic control, Vol.54, No.11, 2009, pp. 2493-2505

[9] S.Chiaverini, "Singularity-Robust Task-Priority Redundancy Resolution for Real-Time Kinematic Control of Robot Manipulators", Int. J. on Robotics an d Automation, Vol.13, No.3, 1997, pp. 398-410

[10] O.Khatib, "A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation", IEEE Journal of Robotics and Automation, Vol.RA-3, No.1, 1987, pp.

[11] O.Khatib, L.Sentis, J.Park and J.Warren, "Whole-Body Dynamic Behavior and Control of Human-like Robots", International Journal of Humanoid Robotics, Vol.1, No.1, 2004, pp. 29-43

[12] L.Sentis and O.Khatib, "Control of Free-Floating Humanoid Robots Through Task Prioritization", In Proc. of the Int. Conf.on the Robotics and Automation, 2005, pp. 1718-1723

[13] E.Yoshida, I.Belousov, C.Esteves and J.P Laumond, "Humanoid Motion Planning for Dynamic Tasks", In Proc. Int. Conf. on Humanoid Robots, 2005, pp. 1-6

[14] N.Mansard and O.Khatib, "Continuous Control Law from Unilateral Constraints : Application to Reactive Obstacle Avoidance in Operational Space", In Proc. Int. Conf. on Robotics and Automation, 2008, pp. 3359-3364

[15] P.Chiacchio, S.Chiaverini, L.Sciavicco and B.Siciliano, "Closed-Loop Inverse Kinematics Schemes for Constrained Redundant Manipulators with Task Space Augmentation and Task Priority Strategy", The Int. J. Robot. Res., Vol.10, No.4, 1991, pp. 410-425

[16] O.khatib, "Real-time obstacle avoidance for manipulators and mobile robots", The Int. J. Robot. Res., Vol.5, No.1, 1986, pp. 90-98

[17] ROBOTICSLAB, http://www.rlab.co.kr/

APPENDIX I
EQUIVALENCE WITH CONTINUOUS INVERSE METHOD

The proposed approach is to modify the desired values for x_2 to smooth the solution out. This solution can be rearranged as follows:

$$\dot{q} = J^{+} \begin{pmatrix} \dot{x}_1 \\ h_2\dot{x}_2 + (1-h_2)J_2J_1^{+}\dot{x}_1 \end{pmatrix} \tag{I.1}$$

$$\dot{q} = J^{+} \begin{pmatrix} I & 0 \\ (1-h_2)J_2J_1^{+} & h_2 I \end{pmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} \tag{I.2}$$

where

$$\begin{aligned} & J^{+} \begin{pmatrix} I & 0 \\ (1-h_2)J_2J_1^{+} & h_2 I \end{pmatrix} \\ &= J^{+} \left[\begin{pmatrix} I & 0 \\ J_2J_1^{+} & 0 \end{pmatrix} + h_2 \begin{pmatrix} 0 & 0 \\ -J_2J_1^{+} & I \end{pmatrix} \right] \\ &= J^{+} \left[\begin{pmatrix} I & 0 \\ J_2J_1^{+} & 0 \end{pmatrix} + h_2 \left[I - \begin{pmatrix} I & 0 \\ J_2J_1^{+} & 0 \end{pmatrix} \right] \right] \\ &= \begin{pmatrix} J_1^{+} & 0 \end{pmatrix} + h_2 \left[J^{+} - \begin{pmatrix} J_1^{+} & 0 \end{pmatrix} \right] \\ &= J^{\oplus H}|_{h_1=1} \end{aligned} \tag{I.3}$$

During the above derivation the following property has been used:

$$JJ_1^{+} = \begin{bmatrix} J_1 \\ J_2 \end{bmatrix} J_1^{+} = \begin{bmatrix} I \\ J_2J_1^{+} \end{bmatrix} \tag{I.4}$$

$$\begin{aligned} \rightarrow J^{+}JJ_1^{+} &= J^{+} \begin{bmatrix} I \\ J_2J_1^{+} \end{bmatrix} \\ J_1^{+} &= J^{+} \begin{bmatrix} I \\ J_2J_1^{+} \end{bmatrix} \end{aligned} \tag{I.5}$$

The operator from $\dot{x}$ to $\dot{q}$ in Equation (I.2) is equivalent to the continuous inverse $J^{\oplus H}$ in [14] when $h_1 = 1$. In a more general case when both h_1 and h_2 are not equal to zero or one, Equation (I.2) becomes

$$\dot{q} = J^{+} \begin{pmatrix} h_1 I & (1-h_1)h_2J_1J_2^{+} \\ (1-h_2)h_1J_2J_1^{+} & h_2 I \end{pmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} \tag{I.6}$$

The equivalence with the continuous pseudo inverse can be easily derived.

$$\begin{aligned} J^{\oplus H} = J^{+} \Bigg\{ & h_1 \begin{pmatrix} I & 0 \\ J_2J_1^{+} & 0 \end{pmatrix} \\ & + h_2 \begin{pmatrix} 0 & J_1J_2^{+} \\ 0 & I \end{pmatrix} - h_1h_2 \begin{pmatrix} 0 & J_1J_2^{+} \\ J_2J_1^{+} & 0 \end{pmatrix} \Bigg\} \end{aligned} \tag{I.7}$$

APPENDIX II
CONTINUITY IN PRIORITY BASED APPROACH

Equation (17), when $h_1 = 0$, is

$$\begin{aligned} \dot{q} &= J_1^{+}J_1J_2^{+}\dot{x}_2 + N_1(J_2N_1)^{+}(\dot{x}_2 - J_2J_1^{+}J_1J_2^{+}\dot{x}_2) \\ &= \{J_1^{+}J_1 + N_1(J_2N_1)^{+}(J_2 - J_2J_1^{+}J_1)\}J_2^{+}\dot{x}_2 \\ &= \{J_1^{+}J_1 + N_1(J_2N_1)^{+}(J_2N_1)\}J_2^{+}\dot{x}_2 \\ &= \{J_1^{+}J_1 + N_1 - N_1 + N_1(J_2N_1)^{+}(J_2N_1)\}J_2^{+}\dot{x}_2 \\ &= \{(J_1^{+}J_1 + N_1) - N_1(I - (J_2N_1)^{+}(J_2N_1))\}J_2^{+}\dot{x}_2 \\ &= \{I - N_1N_{2|1}\}J_2^{+}\dot{x}_2 \\ &= J_2^{+}\dot{x}_2 \end{aligned} \tag{II.1}$$

When $h_1 \neq 0$,

$$\dot{q} = J_2^{+}\dot{x}_2 + h_1\{J_1^{+} - N_1(J_2N_1)^{+}J_2J_1^{+}\}(\dot{x}_1 - J_1J_2^{+}\dot{x}_2) \tag{II.2}$$