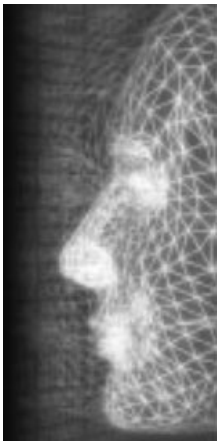
COMPUTER ANIMATION AND VIRTUAL WORLDS
Comp. Anim. Virtual Worlds 2009; **20**: 511–522
Published online 13 February 2009 in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/cav.280

Motion autonomy for humanoids: experiments on HRP-2 No. 14



By Eiichi Yoshida, Jean-Paul Laumond, Claudia Esteves, Oussama Kanoun, Anthony Mallet, Takeshi Sakaguchi and Kazuhito Yokoi*

This paper deals with whole-body motion planning and dynamic control for humanoid from two aspects: locomotion including manipulation and reaching. In the first part, we address a problem of simultaneous locomotion and manipulation planning that combines a geometric and kinematic motion planner with a dynamic humanoid motion generator. The second part deals with whole-body reaching tasks by using a generalized inverse kinematics (IK) method to fully exploit the high redundancy of the humanoid robot. Through experiments using humanoid platform HRP-2 No. 14 installed at LAAS-CNRS, we first verify the validity of each method. An integrated experiment is then presented that unifies the both results via visual perception to execute an object-fetching task. Copyright © 2009 John Wiley & Sons, Ltd.

Received: 18 November 2008; Accepted: 28 November 2008

KEY WORDS: motion planning; humanoid; whole-body motoin; locomotion

Introduction

With their high mobility and high redundancy, humanoid robots are expected to perform complicated tasks. Their anthropomorphic configuration gives another advantage that they can easily adapt to machines or environments designed for humans. Recent progress in hardware accelerates diverse research in humanoid robots. Various types of tasks have been performed: manipulation[1,2] or serving tasks.[3,4] One of the key issues to fully exploit the capacity of humanoid robots is to develop a methodology that enables them to execute various tasks requiring dynamic and smooth whole-body motions including collision avoidance and locomotion, like an object carrying task.

In the field of motion planning, recent advancement in probabilistic methods has greatly improved the three-dimensional (3D) motion planning for mechanism involving complicated geometry and many degrees of freedom (e.g., Reference [5]). However, most of those methods are based on the geometric and kinematic planning in the configuration space whereas tasks for humanoid robots are often specified in the workspace and subject to dynamics including balance keeping.

Concerning control issues of humanoid robots, powerful controllers have been developed to generate whole-body dynamic motion in a reactive manner (e.g., Reference [6]). As for locomotion, stable motion pattern can be generated efficiently thanks to the progress in biped walking control theory, mainly based on zero moment point (ZMP) control (e.g., Reference [7]). Planning of 3D humanoid motion for tasks in complex environments has to benefit from these two domains.

This paper addresses two aspects of humanoid whole-body motion, simultaneous planning of locomotion and manipulation and also dynamic reaching. In the first part of this paper, we propose a two-stage planning framework based on the geometrical and kinematic planning technique whose output is validated by dynamic motion pattern generator. The second part addresses how to exploit the high redundancy of humanoid robots when performing reaching or grasping tasks. The last part of the paper presents an integrated experiment with the humanoid robot platform HRP-2

*Correspondence to: Eiichi Yoshida, Intelligent Systems Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), 1-1-1 Umezono, Tsukuba 305-8568, Japan. E-mail: e.yoshida@aist.go.jp

No. 14 to apply the proposed methods to a task in the real world through visual perception.

Manipulating While Walking

Humanoid motion planning is becoming a hot topic since it faces complexity of planning and dynamic control at the same time. Kuffner *et al*. proposed various types of humanoid motion planners,[8–10] such as balancing, footstep planning, and navigation, while displacing movable obstacles.

Okada *et al*. addressed the motion planning for collision-free whole-body posture control[11] by dividing the robot into movable, fixed and free limbs using RRT planner. Yoshida proposed humanoid motion planning based on multi-level DOF exploitation.[12]

In the domain of computer graphics, motion editing is an active area of research. Gleicher classified various constraint-based methods that take into account spatial and temporal constraints, which often correspond to the problems of inverse kinematics (IK) and filtering, respectively.[13] In the field of graphic animation of digital actors, recent progress in randomized motion planning is currently being actively applied.[5,14]

Two-stage Planning Method

In this section, we summarize the two-stage planning method, we have proposed in Reference [15] as illustrated in Figure 1. At the first stage, the motion planner computes a collision-free walking path for the lower part of the robot body approximated by a bounding box, as well as a collision-free path for the upper body. In the second stage, this output path is given as inputs to the dynamic pattern generator[7] of the humanoid robot that transforms it into a dynamically executable motion. The joint angle command for the whole body is computed by taking account of dynamic balance based on ZMP. If the generated dynamic motion induces unpredicted collisions due to deviation from the geometrically and kinematically planned path, then the planner goes back to the first stage to "reshape" the previous path as explained in the next section.

Smooth Motion Reshaping

A collision-free path issued from the first motion planning stage, will not always result in a collision-free

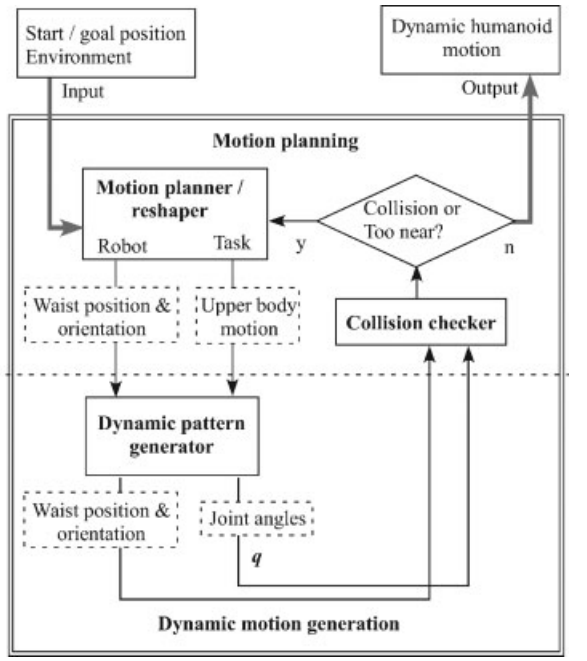


Figure 1. Two-stage motion planning framework.

trajectory after dynamic pattern generation is performed. If the variation of the motion is small enough, those collisions will be with the humanoid's upper body or the carried object. In such a case, we can assume that local reshaping of the trajectory will suffice to avoid the obstacles without replanning the whole nominal trajectory.

When a collision is found, a new random collision-free configuration near the colliding one is first generated, and then an IK solver is applied to ensure the geometric constraints of the end-effector. Although collision-free motions can be generated at that stage, lack of smoothness in velocity profile might cause instability or unnecessary oscillation when it is executed by the humanoid robot. Then we propose a reshaping method that accounts for the smoothness of the motion when avoiding the obstacles.

The reshaping procedure is performed in the following two steps illustrated in Figure 2 by accounting for motion continuity:[15]

1. A smooth trajectory to be followed by the end-effector is specified in the task space and resampled at each sampling time (5 millisecond) to enforce temporal constraints (Figure 2(a)–(c)).
2. An IK specified the motion of the end-effector enforcing geometric constraints (Figure 2(d)).

DOI: 10.1002/cav

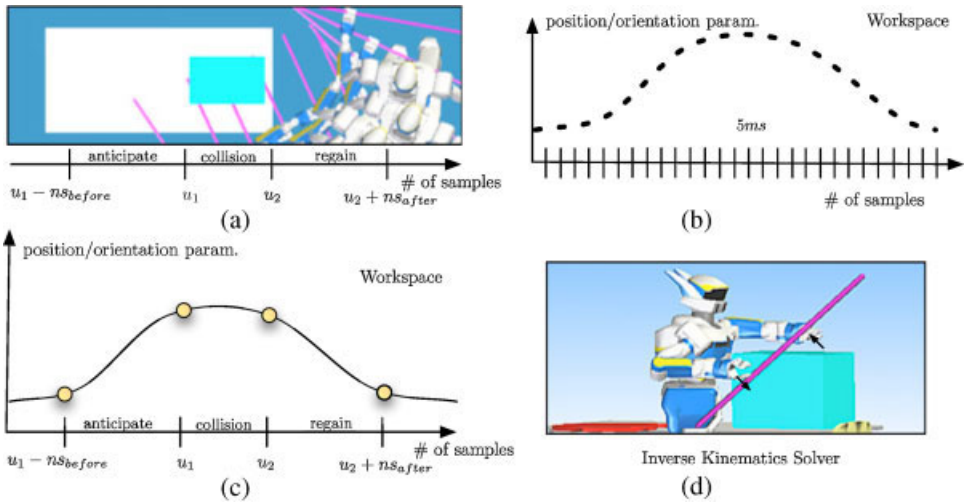


Figure 2. (a) A top view of the volume swept by the robot avoiding a box on the table. Collisions occur between the bar and the box. The reshaping limits are set by identifying the anticipating, colliding and regaining times. (b) Smooth motion is specified in the task space by interpolating the bar's configuration at key frames. (c) The bar's motion is resampled at 5 millisecond to replace its original motion. (d) New constraints are enforced by using a whole-body IK solver.

Experiments with HRP-2 No. 14

We have conducted experiments of the proposed humanoid motion planner using the simulator OpenHRP[16] and the hardware humanoid platform HRP-2 No. 14 installed at LAAS-CNRS. HRP-2 has 30 degrees of freedom with 1.54 m in height and 58 kg in weight.[17] This robot has two chest joints for pitch and yaw rotation, which extends the motion capability including lying down on the floor and standing up. We take an example of a task carrying a bar in an environment populated by obstacles. The length, diameter and weight of the bar is 1.8 m, 2.4 cm, and 0.5 kg, respectively. Figure 3 illustrates a real experiment. After the robot started walking, it lifted the bar to move it by avoiding the collision with the box on the table (Figure 3(b)–(d)). The bar is lowered to the initial height after collision avoidance (Figure 3(e) and (f)) to reach the goal position. The dynamic task has successfully been achieved, which validates the proposed planner.

Task-driven Support Polygon Reshaping for Reaching

We address a task-driven motion generation method that allows a humanoid robot to make whole-body motions including support polygon reshaping to achieve the given tasks.[18] There are many works in the literature that have focused on the generation of whole-body motions for complex mechanisms such as humanoid robots or digital actors. A popular approach for motion specification has been, instead of setting explicitly the value of all degrees of freedom, to only specify the values of a task to be accomplished by the end-effector. The idea is to benefit from the redundancy of the mechanism to choose the solution that best solves the task according to some constraints. Among these works, generalized IK algorithms that project tasks with lower priority into the null space of the Jacobian of the higher priority tasks have been widely studied (e.g., References [6,19–22]).

Our contribution is to consider the possibility of reshaping the support polygon by stepping to increase the accessible space of the end-effectors in the 3D space. The problem we address can be viewed as a 3D extension of the 2D problem addressed in Reference [23]. In Reference [23], the authors propose a strategy for the control of a pattern generator by monitoring the arm manipulability. While their model lies in the sagittal plane, our approach makes use of the whole body motion in 3D space. Moreover, in spite of our reasoning being based on IK and simple geometric support polygon reshaping, our method guarantees that the motion is dynamically stable. This property is a consequence of the pattern generator,[7] we use to generate the stepping behavior.

DOI: 10.1002/cav

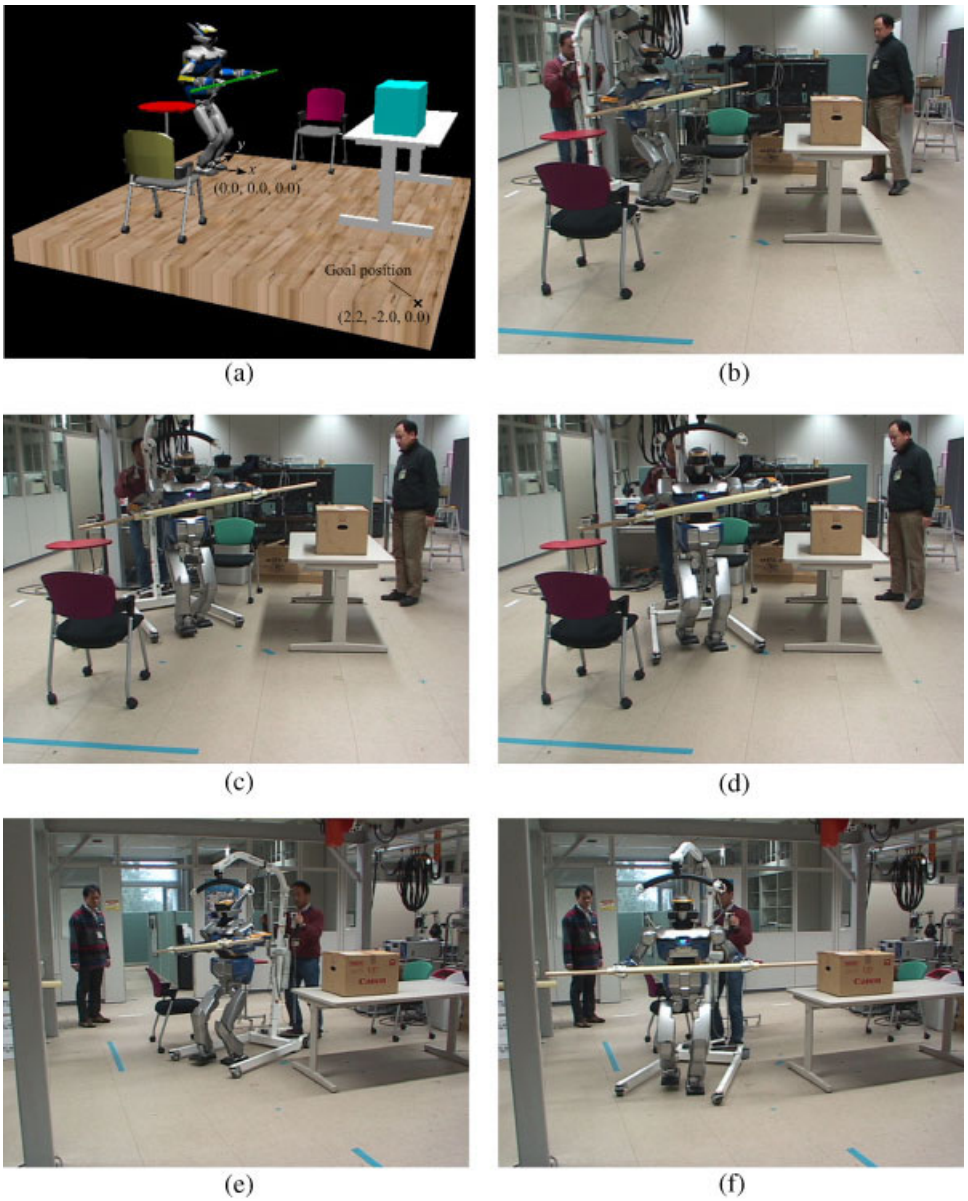


Figure 3. Experiment of planned bar-carrying task. (a) Initial configuration, (b) starts walking, (c) starts lifting the bar, (d) the bar passes above the obstacle, (e) lowering the bar after avoiding collision, and (f) going to final position.

Method Overview

The support polygon reshaping integrates two important components, the generalized IK and dynamic walking pattern generator. Figure 4 shows an overview of the method. The task is specified in the workspace as a desired velocity $\dot{\boldsymbol{x}}_j$ with priority j from which the generalized IK solver computes the whole-body motion as joint velocities $\dot{\boldsymbol{q}}$ of the robot. Meanwhile, several criteria such as manipulability or joint limit are monitored.

As long as those criteria are satisfied, the computation of whole-body motion continues until the target of the task is achieved. If the task cannot be achieved due to unsatisfied criteria, the support polygon planner is triggered in order to extend reachable space. A geometric module determines the direction and position of the deformation of support polygon so that the incomplete

DOI: 10.1002/cav

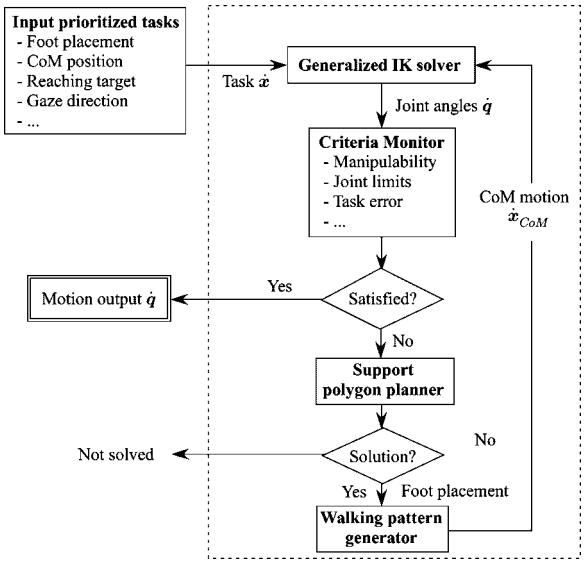


Figure 4. Task-driven support polygon reshaping and whole-body motion generation.

task is fulfilled. The position of a foot is then derived to generate the motion of center of mass (CoM) $\dot{\boldsymbol{x}}_{\mathrm{CoM}}$ by using a dynamic walking pattern generator.[7] Using this CoM motion, the original task is then redefined as the whole-body motion including stepping that is recalculated using the same generalized IK solver.

Let us first overview the generalized IK framework. Then we will show how the support polygon is reshaped.

Generalized Inverse Kinematics for Whole-body Motion

Inverse Kinematics for Prioritized Tasks. Let us consider a task $\dot{\boldsymbol{x}}_j$ with priority j in the workspace and the relationship between the joint angle velocity $\dot{\boldsymbol{q}}$ is described using Jacobian matrix, like $\dot{\boldsymbol{x}}_j = \boldsymbol{J}_j\dot{\boldsymbol{q}}$. For the tasks with the first priority, using pseudoinverse $\boldsymbol{J}_1^{\#}$, the joint angles that achieves the task is given

$$\dot{\boldsymbol{q}}_1 = \boldsymbol{J}_1^{\#}\dot{\boldsymbol{x}}_1 + (\boldsymbol{I}_n - \boldsymbol{J}_1^{\#}\boldsymbol{J}_1)\boldsymbol{y}_1 \tag{1}$$

where $\boldsymbol{y}_1$, n, and $\boldsymbol{I}_n$ are an arbitrary vector, the number of the joints and identity matrix of dimension n, respectively.

For the task with second priority $\dot{\boldsymbol{x}}_2$, the joint velocities $\dot{\boldsymbol{q}}_2$ is calculated as follows:[19]

$$\begin{aligned}\dot{\boldsymbol{q}}_2 &= \dot{\boldsymbol{q}}_1 + \hat{\boldsymbol{J}}_2^{\#}(\dot{\boldsymbol{x}}_2 - \boldsymbol{J}_2\dot{\boldsymbol{q}}_1) + (\boldsymbol{I}_n - \boldsymbol{J}_1^{\#}\boldsymbol{J}_1)(\boldsymbol{I}_n - \hat{\boldsymbol{J}}_2^{\#}\hat{\boldsymbol{J}}_2)\boldsymbol{y}_2 \\ \text{where} \quad \hat{\boldsymbol{J}}_2 &\equiv \boldsymbol{J}_2(\boldsymbol{I}_n - \boldsymbol{J}_1^{\#}\boldsymbol{J}_1)\end{aligned} \tag{2}$$

where $\boldsymbol{y}_2$ is an arbitrary vector of dimension n. It can be extended to the task of jth ($j \geq 2$) priority in the following formula:[20,21]

$$\begin{aligned}\dot{\boldsymbol{q}}_j &= \dot{\boldsymbol{q}}_{j-1} + \hat{\boldsymbol{J}}_j^{\#}(\dot{\boldsymbol{x}}_j - \boldsymbol{J}_j\dot{\boldsymbol{q}}_{j-1}) + \boldsymbol{N}_j\boldsymbol{y}_j \\ \boldsymbol{N}_j &\equiv \boldsymbol{N}_{j-1}(\boldsymbol{I}_n - \hat{\boldsymbol{J}}_j^{\#}\hat{\boldsymbol{J}}_j), \;\; \hat{\boldsymbol{J}}_j \equiv \boldsymbol{J}_j(\boldsymbol{I}_n - \hat{\boldsymbol{J}}_{j-1}^{\#}\hat{\boldsymbol{J}}_{j-1})\end{aligned} \tag{3}$$

Weighted Pseudoinverse. In most cases, it is preferable for a humanoid robot to use the lighter links to achieve tasks. For this purpose, we introduce a weighted pseudoinverse:

$$\boldsymbol{J}_W^{\#} = (\boldsymbol{J}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{J})^{-1}\boldsymbol{J}^{\mathrm{T}}\boldsymbol{W}, \;\; \boldsymbol{W} = \mathrm{diag}\{\sqrt{W_1}, \ldots \sqrt{W_n}\} \tag{4}$$

The weight W_i of each joint is given as the ratio of the mass m_i of the link i to the total mass M, namely m_i/M. Moreover, a selection matrix $\boldsymbol{S} = \mathrm{diag}\{S_1, \ldots S_n\}$ ($S_i = 0$ or 1) is multiplied to this inverse to select the activated joints according to the task specification. The selection matrix is set to $\boldsymbol{I}_n$ if all the joints are used to achieve the task.

Using this weighted Jacobian first lighter links are used then heavier ones. By combining a selection matrix $\boldsymbol{S}_l$ that forbids using the joints approaching the limit of the movable range, the heuristics of whole-body motion workspace extension[22] can be implemented in a simpler way.

Monitoring Task Execution Criteria. While the motion is being computed by the generalized IK, several properties are monitored.

One of the important measures is the manipulability[24] defined as

$$w \equiv \sqrt{\det\{\boldsymbol{J}\boldsymbol{J}^{\mathrm{T}}\}} \tag{5}$$

This measure is continuously tracked during the motion generation as well as others such as joint angle limits or end-effector errors from the target. If it becomes below a certain value, it means that it is difficult to achieve the task.

Joint limit constraints can be taken into account by introducing another selection diagonal matrix $\boldsymbol{S}_l$ whose ith component become zero if the corresponding joint reaches a limit angle.

As shown in Figure 4, when one or more monitored measures go out of the admissible range to prevent the task from being achieved, the support polygon reshaping

DOI: 10.1002/cav

is launched to extend the accessible space as detailed in the next subsection.

Support Polygon Reshaping

Figure 5 shows the proposed support polygon reshaping scheme. This simple algorithm allows the humanoid robot to make a step motion, keeping a large margin of accessible area for the task by facing the upper body to the target direction.

Then the CoM motion $\dot{\boldsymbol{x}}_{\mathrm{CoM}}$ is computed from the new foot position by the walking pattern generator based on the preview control of ZMP.[7] The basic idea is to calculate the CoM motion by anticipating the desired future ZMP positions derived from the footsteps.

Finally the original task is redefined as another problem of whole-body task using this newly generated CoM motion with an additional task of CoM, which is represented by CoM Jacobian.[25] The same generalized IK solver framework is used to incorporate the motion required for the task and the stepping motion in the whole-body level.

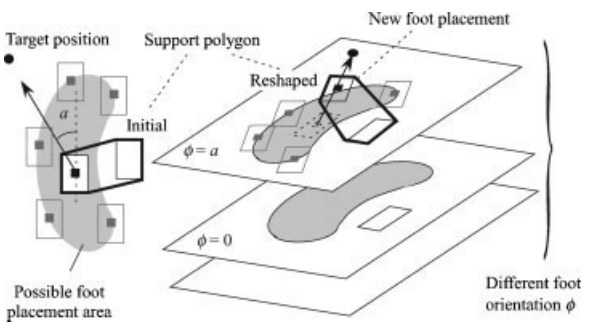


Figure 5. Support polygon reshaping method.

Experimental Results

In the following experiment, the humanoid robot is required to reach a position with the left hand. Four tasks are given with the following priority (i) foot placement, (ii) CoM position, (iii) hand reaching task, and (iv) gaze direction in the order of higher priority. For all the tasks the weighted Jacobian (4) is utilized for IK. As for the selection matrix $\boldsymbol{S}$, all the degrees of freedom are used, namely setting $\boldsymbol{S}$ to $\boldsymbol{I}_n$, for all the tasks. The reaching task is defined by the target positions without specifying orientation of the hand.

The monitored criteria here during the motion are the manipulability of the arm and the error between the reference end-effector position and the one calculated by the IK solver. The robot tries to reach the target first with the CoM position at the center of the initial support polygon. If those values go below a certain threshold, the support polygon reshaping process is activated. Here the thresholds of manipulability and end-effector error are empirically set to 1.5×10^{-4} and 4.0×10^{-5} m, respectively.

Figure 6 shows the snapshots of a reaching task including reshaping. The manipulability measure for this task is given in Figure 7 to compare to the motion without reshaping. Without reshaping, the arm would approach a singular configuration where the manipulability becomes lower than the threshold at 2.6 second. The computation keeping the same support polygon is then discarded. The reshaping starts at this moment to recalculate the overall whole-body motion including stepping. We can see the manipulability regains higher value at the final position.

In Figure 8, the time development of x and y positions of ZMP measured from the ankle force sensors are plotted for the sideways reaching motion. The dotted and solid lines are the planned and measured trajectories, respectively. The shaded areas in those graphs depict the

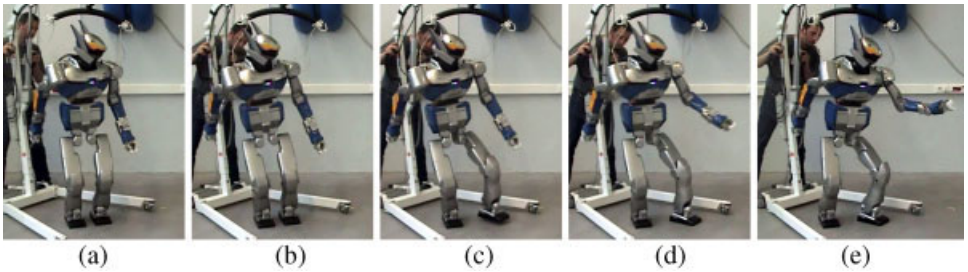


Figure 6. Experimentation on HRP-2. Putting weight on the right foot in (b), the robot goes through a posture that is not statically stable (c) to finish stepping in (c). The final goal of the end effector is achieved at (e). Notice that the robot makes a whole-body motion including reaching task, stepping and keeping the gaze.

DOI: 10.1002/cav

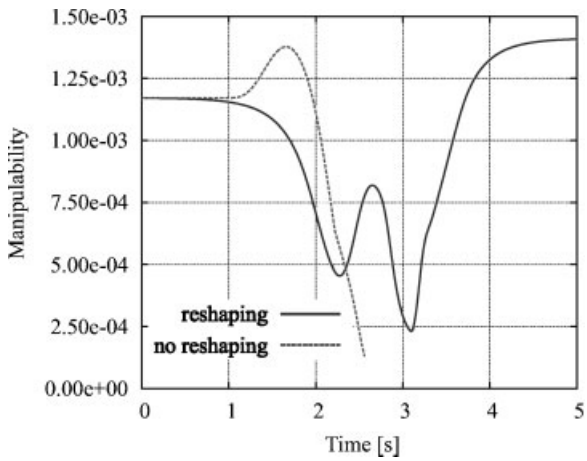


Figure 7. Without support polygon reshaping, the manipulability measure decreases below the threshold. Although it also decreases with reshaping, the manipulability increases in the course of stepping motion.

transition of support polygon area projected on x- and y-axis. As we can see, the planned trajectories of ZMP always stay inside the support polygon. Note that the final ZMP position in x direction goes out of the initial support polygon: this means the reaching task could not have been performed without stepping.

Motion in Real World: Integrating With Perception

The presented motion planning methods are currently integrated with perception, principally vision, to make actions in the real world. This integration allows the robot to execute such commands as "go to the yellow table" and "take the orange ball."

Object Recognition and Localization

The HRP-2 robot is equipped with two pairs of firewire digital color cameras, configured as two independent stereo-vision camera pairs. We here utilize standard state of the art components to implement a simple function of object recognition and localization.

For the detection, the model of the objects to be detected are previously learned using two dimensional histogram in the $\{Hue, Saturation\}$ color space by taking a sample image with a color space. The object detection is performed by *back projecting* the object histogram onto a video image. The back projection image is obtained by replacing each pixel value by the corresponding value in the model histogram (Figure 9). We use a method called *Continuously Adaptive Mean SHIFT (CAMSHIFT)* algorithm[26] to locate the object center and orientation in the back projection image.

A stereo-vision algorithm by pixel correlation is applied on the stereo image pairs, and produces a dense 3D image of the current scene. Even though pixel

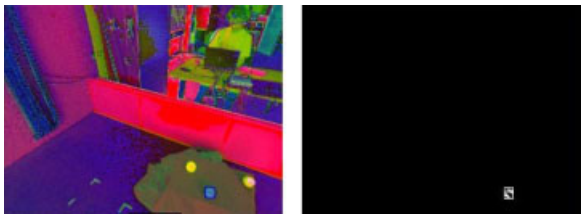


Figure 9. Object detection. Left image shows the HSV image and right image is the back projection of the table color model in the source image. Rectangle is the result of the execution of the CAMSHIFT algorithm on the back projection image.

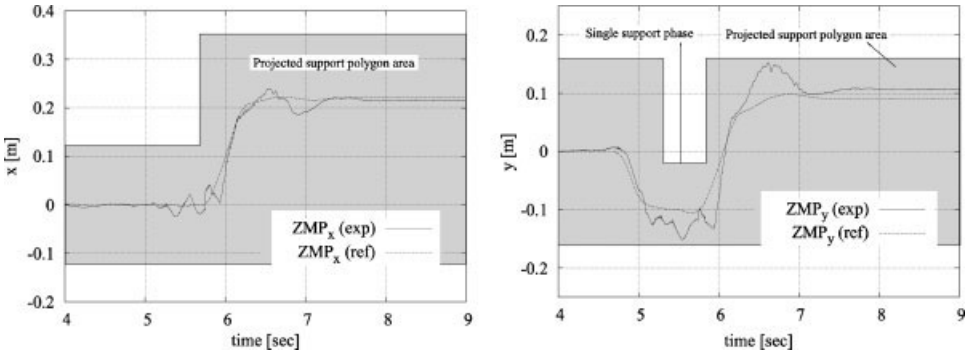


Figure 8. Evolution of the ZMP coordinates during the motion. The shaded area expresses the transition of the projection of support polygon in x-axis (left) et in y-axis (right).

DOI: 10.1002/cav

correlation is known to give poor results in indoor environments, the objects to localize are sufficiently textured so that precise enough 3D points can be obtained in the vicinity of the objects.

Coupling the Motion Planner with Perception

The motion planners presented in previous sections are integrated with the vision system so that the robot can execute a task composed of navigation and object grasping.

For navigation, we apply the same type of two-stage motion planner for navigation planning presented in the second section. At the first stage, a collision-free smooth locomotion path is calculated for the approximated bounding box. It is desirable for the robot to walk forward in order to look at the object and to take it. This preference can be modeled as a nonholonomic constraint, and we can benefit from well-developed planning method of a smooth path for car-like robot.[27] Then the path is transformed into dynamic humanoid locomotion at the second stage by applying the dynamic walking patter generator. This navigation planner allows the humanoid robot to go in front of the visually located colored table several meters away by avoiding known obstacles as shown in Figure 10.

The whole-body motion generator presented in the third section is used for the grasping task. Given the object location from the vision system, the whole-body motion generator computes automatically a reaching motion, including stepping depending on the detected object location.

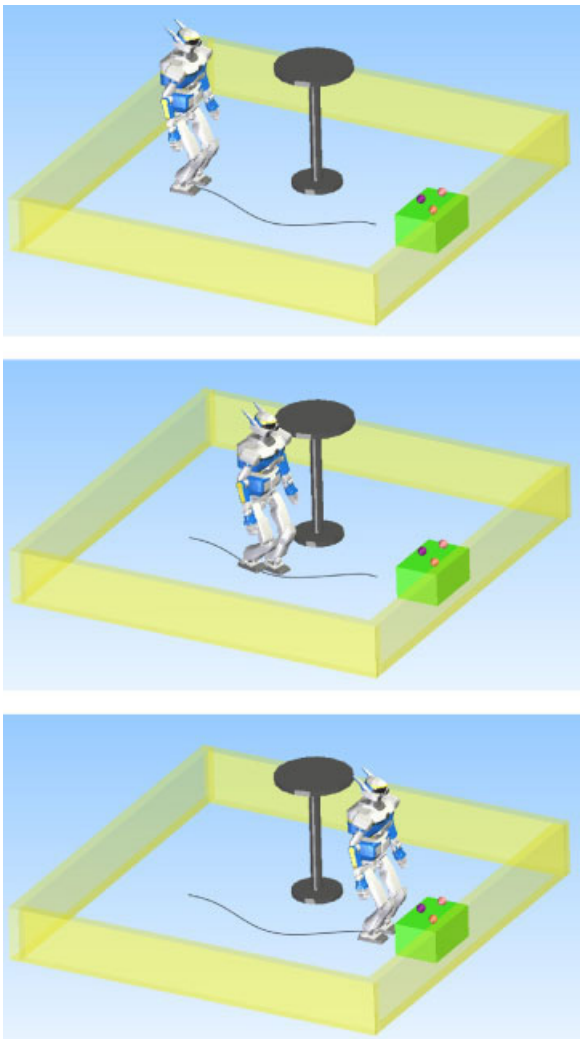


Figure 10. A planned smooth walking trajectory to a target position.

Experiments

We have conducted experiments to validate the integrated system. The humanoid robot is given a task to take a colored ball and put it at another place. The task is decomposed into several generic action commands, such as detection and localization of a learned object, locomotion to a location, and hand reaching to a position in 3D, with other simple tasks like turning on the spot and gripper opening and closing.

A simple supervision system that can invoke the actions with scripts is utilized to manage the robot behavior easily. Each action can report failures (e.g., failure in grasping an object). It is thus possible to implement error recovery strategies by analyzing the reports of the actions. In the following experiment, each action is associated with a vocal command to allow the user to give a sequence of commands to the robot in an interactive manner.

Figure 11 shows snapshots of experiments. Since the ball is too far away to be detected with camera at the initial position, the humanoid robot first localizes the green box on which the balls are placed (Figure 11a). The robot walks with a smooth trajectory in front of the box (Figure 11b) and localizes precisely the colored ball to grasp (Figure 11c). Then the whole-body reaching motion is executed to grasp the ball (Figure 11d). After turning, the robot is told to detect a colored table and walks toward it always with a smooth trajectory (Figure 11e).

DOI: 10.1002/cav

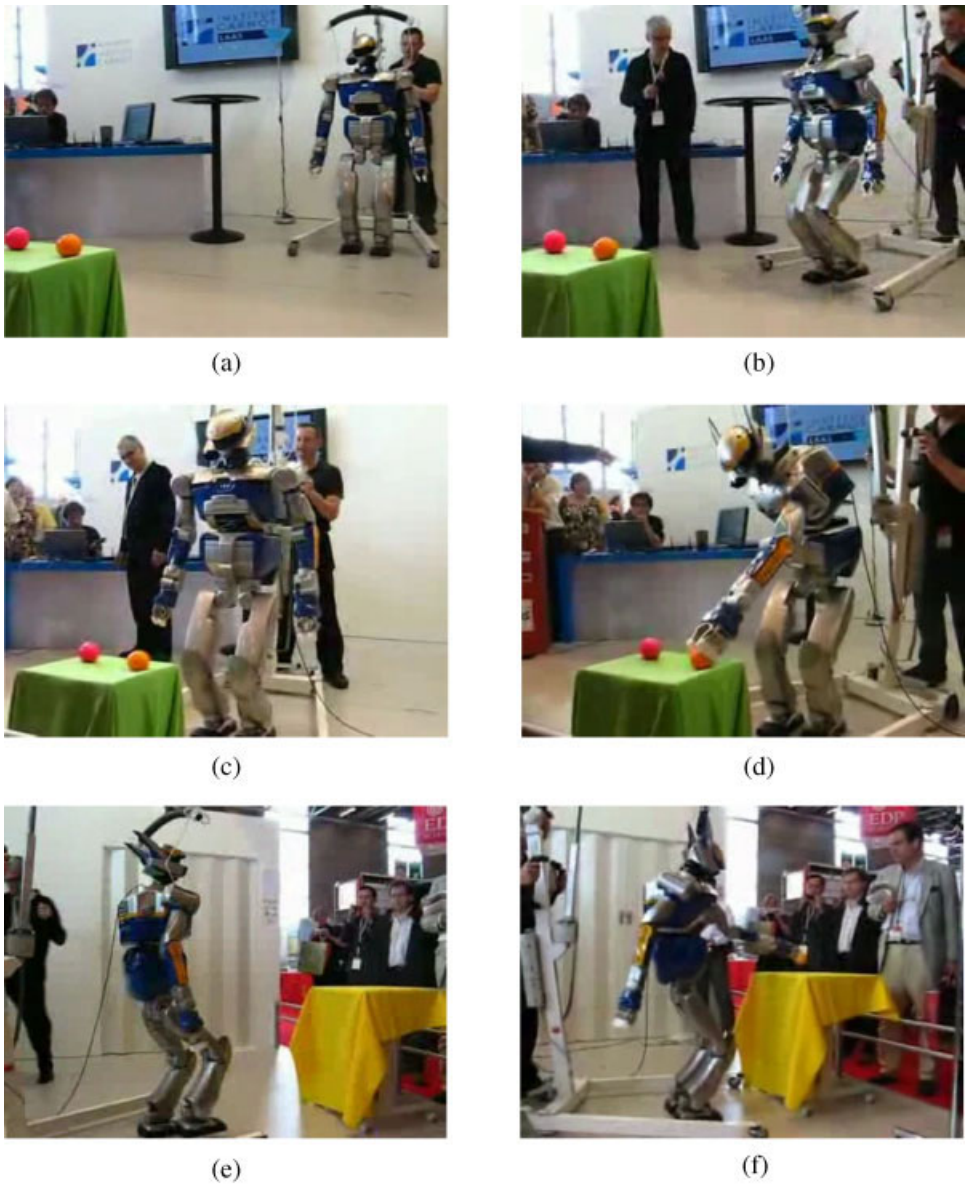


Figure 11. Ball-fetching task using visual perception and motion planner. (a) Localization of the box, (b) walking to the box, (c) detection and localization of the ball, (d) whole-body reaching for grasping, (e) locomotion to another location, and (f) putting the ball on the detected table.

Finally it puts the ball on the table again with whole-body motion (Figure 11f).

This experiment was conducted more than 10 times in an exposition in front of the public using vocal interaction by a human operator. Since the location of the robots and objects are different at every demonstration, it happened that the robot failed to grasp with unexpected disturbances or localization errors. However, the task could be executed again successfully thanks to the generality of the action commands, by just repeating the same action command. As a result, all the demos were successful including those retries. This validates the reliability of the proposed motion planner, the integrated perception system and also the robustness of task execution framework.

DOI: 10.1002/cav

Conclusions

The goal of this paper is to present our work in progress on the motion autonomy in humanoid robotics. Even though Robotics and Computer Animation follow different goals with respect to their respective application fields, we believe that the research in both areas should benefit from a synergetic point of view.

The synergy possibly comes from a common objective aiming at better understanding the computational issues of human motions. The questions addressed in this paper (How to combine manipulation and locomotion tasks? How to enlarge the scope of redundant system based methods?) are generic questions challenging for both servicing robotics and game industry.

Another potential synergy concerns the physical interaction. We believe Robotics can contribute to Computer Animation domain with our feedback from the real-world experiments. Such contributions include planning dynamically plausible motion for digital actors and also the physical interaction with virtual environment, which can be applied to interactive game or animations.

ACKNOWLEDGEMENTS

This research was partially supported by Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research (B), 18300070, 2006.

References

1. Harada H, Kajita S, Saito H, *et al*. A humanoid robot carrying a heavy object. In *Proceedings of 2005 IEEE International Conference on Robotics and Automation*, 2005; 1712–1717.
2. Yoshida E, Poirier M, Laumond J-P, *et al*. Whole-body motion planning for pivoting based manipulation by humanoids. In *Proceedings of 2008 IEEE International Conference on Robotics and Automation*, 2008; 1712–1717.
3. Okada K, Kojima M, Sagawa Y, Ichino T, Sato K, Inaba M. Vision based behavior verification system of humanoid robot for daily environment tasks. In *Proceedings of 2006 IEEE-RAS International Conference on Humanoid Robots*, 2006; 7–12.
4. Neo ES, Sakaguchi T, Yokoi K, Kawai Y, Maruyama K. A behavior level operation system for humanoid robots. In *Proceedings of 2006 IEEE-RAS International Conference on Humanoid Robots*, 2006; 327–332.
5. Esteves C, Arechavaleta G, Pettré J, Laumond J-P. Animation planning for virtual characters cooperation. *ACM Transactions on Graphics* 2006; **25**(2): 319–339.
6. Sentis L, Khatib O. Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. *International Journal of Humanoid Robotics* 2005; **2**(4): 505–518.
7. Kajita S, Kanehiro F, Kaneko K, *et al*. Biped walking pattern generation by using preview control of zero-moment point. In *Proceedings of 2003 IEEE International Conference on Robotics and Automation*, 2003; 1620–1626.
8. Kuffner J, Kagami S, Nishiwaki K, Inaba M, Inoue H. Dynamically-stable motion planning for humanoid robots. *Autonomous Robots* 2002; **12**(1): 105–118.
9. Chestnutt J, Kuffner J. A tiered planning strategy for biped navigation. In *Proceedings of 2004 IEEE-RAS International Conference on Humanoid Robotics*, 2004; 422–436.
10. Stilman M, Kuffner J. Navigation among movable obstacles: real-time reasoning in complex environments. In *Proceedings of 2004 IEEE-RAS International Conference on Humanoid Robotics*, 2004; 322–341.
11. Okada K, Ogura T, Haneda A, *et al*. Integrated system software for HRP humanoid. In *Proceedings of 2004 IEEE International Conference on Robotics and Automation*, 2004; 3207–3212.
12. Yoshida E. Humanoid motion planning using multi-level dof exploitation based on randomized method. In *Proceedings of 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005; 25–30.
13. Gleicher M. Comparing constraint-based motion editing method. *Graphical Models* 2001; **63**: 107–134.
14. Yamane K, Kuffner J, Hodgins J. Synthesizing animations of human manipulation tasks. *ACM Transactions on Graphics* 2004; **23**(3): 532–539.
15. Yoshida E, Esteves C, Sakaguchi T, Yokoi K, Laumond J-P. Smooth collision avoidance: practical issues in dynamic humanoid motion. In *Proceedings of 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006; 208–213.
16. Kanehiro F, Hirukawa H, Kajita S. Openhrp: open architecture humanoid robotics platform. *International Journal of Robotics Research* 2004; **23**(2): 155–165.
17. Kaneko K, Kanehiro F, Kajita S, *et al*. The humanoid robot hrp-2. In *Proceedings of 2004 IEEE International Conference on Robotics and Automation*, 2004; 1083–1090.
18. Yoshida E, Kanoun O, Esteves C, Laumond J-P, Yokoi K. Task-driven support polygon reshaping for humanoids. In *Proceedings of 2006 IEEE-RAS International Conference on Humanoid Robots*, 2006; 827–832.
19. Nakamura Y. *Advanced Robotics: Redundancy and Optimization*. Addison-Wesley Longman Publishing: Boston, 1991.
20. Siciliano B, Slotine J-JE. A general framework for managing multiple tasks in highly redundant robotic systems. In *Proceedings of IEEE International Conference on Advanced Robotics*, 1991; 1211–1216.
21. Baerlocher P, Boulic R. An inverse kinematics architecture enforcing and arbitrary number of strict priority levels. *The Visual Computer* 2004; **20**: 402–417.
22. Sian NE, Yokoi K, Kajita S, Tanie K. A framework for remote execution of whole body motions for humanoid robots. In *Proceedings of 2004 IEEE/RAS International Conference on Humanoid Robots*, 2004; 608–626.
23. Yoshida H, Inoue K, Arai T, Mae Y. Mobile manipulation of humanoid robots—a method of adjusting leg motion for improvement of arm's manipulability. In *Proceedings of 2001*

DOI: 10.1002/cav

IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2001; 266–271.
24. Yoshikawa T. Manipulability of robotic mechanisms. *International Journal on Robotics Research* 1985; **4**(2): 3–9.
25. Sugihara T, Nakamura Y, Inoue H. Realtime humanoid motion generation through zmp manipulation based on inverted pendulum control. In *Proceedings of 2002 IEEE International Conference on Robotics and Automation*, 2002; 1404–1409.
26. Bradski GR. Computer video face tracking for use in a perceptual user interface. *Intel Technology Journal*, 1998. http://download.intel.com/technology/itj/q21998/pdf/camshift.pdf
27. Laumond J-P (ed.). *Robot Motion Planning and Control*, Vol. 229 of Lectures Notes in Control and Information Sciences. Springer, 1998.

Authors' biographies:

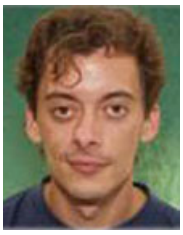
Eiichi Yoshida received ME and PhD degrees on Precision Machinery Engineering from Graduate School of Engineering, the University of Tokyo in 1993 and 1996, respectively. In 1996, he joined former Mechanical Engineering Laboratory, Tsukuba, Japan. He is currently senior research scientist, in Intelligent Systems Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan. From 1990 to 1991, he was visiting research associate at Swiss Federal Institute of Technology at Lausanne (EPFL). He currently serves as Co-Director of AIST/IS-CNRS/ST2I Joint French-Japanese Robotics Laboratory (JRL) at LAAS-CNRS, Toulouse, France. His research interests include robot task and motion planning, modular robotic systems, and humanoid robots.

Jean-Paul Laumond is Directeur de Recherche at LAAS-CNRS in Toulouse, France, where he is also associated with AIST/IS-CNRS/ST2I Joint French-Japanese Robotics Laboratory (JRL), as its Co-Director. In Fall 1990, he was invited to be a senior scientist at Stanford University. He has been a member of the French Comité National de la Recherche Scientifique from 1991 to 1995. He is currently a member of the board of the ACI Neurosciences Intégratives et Computationnelles. He has been coordinator of two European Esprit projects-PROMotion (1992–1995) and MOLOG (1999–2002)–both dedicated to robot motion planning technology. In 2001 and 2002 he created and managed Kineo CAM, a spin-off company from LAAS-CNRS devoted to developing and marketing motion planning technology. Kineo CAM was awarded the French Research Ministery prize for innovation and enterprise in 2000. He teaches robotics at ENSTA and Ecole Normale Supérieure in Paris. He has edited three books and published more than 100 papers in international journals and conferences in computer science, automatic control and robotics. His current research interests include human motion studies along three perspectives: artificial motion for humanoid robots, virtual motion for digital actors and mannequins, and natural motions of human beings.

Claudia Esteves received her PhD in Computer Science from the National Polytechnic Institute of Toulouse in 2007. Her PhD research was conducted at LAAS-CNRS under the supervision of J-P. Laumond on Motion Planning for Virtual Characters and Humanoid Robots. She is currently an Associate Professor in the Faculty of Mathematics at the University of Guanajuato, Mexico. Her research interests include mainly motion planning for human-like mechanisms and motion planning with constraints.

Oussama Kanoun graduated from Ecole des Mines de Paris with a major in Robotics. Obtained a Master degree from ENS Cachan in Mathematics for Computer Vision and Learning. Currently PhD student at LAAS-CNRS.

DOI: 10.1002/cav



Anthony Mallet has graduated from the Ecole Centrale de Lyon, France, in 1996. He received the PhD degree in computer science from the Institut National Polytechnique de Toulouse, France, in 2001 for his research on localization of field robots at LAAS-CNRS. He was with the ASL lab of the EPFL, Lausanne, Switzerland, for one year as a Postdoctoral fellow and worked for two years in a private company as a software research and innovation engineer. He is currently a research engineer with LAAS-CNRS, Toulouse, France, and works on humanoid robots software architecture.

Takeshi Sakaguchi received the BSc degree in control engineering, MSc degree in the design and the control of the D.D. robot, and the PhD degree in dynamic manipulation dealing with moving objects from Osaka University, Osaka, Japan, in 1987, 1989 and 1993, respectively. He is currently a senior researcher in the AIST/IS-CNRS/ST2I Joint French-Japanese Robotics Laboratory (JRL), the National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan. His current research interests include ubiquitous robotics, ambient intelligence, and interactive human-space design and intelligence.

Kazuhito Yokoi received his BE degree in Mechanical Engineering from Nagoya Institute of Technology in 1984, and the M E and PhD degrees Mechanical Engineering Science from the Tokyo Institute of Technology 1986 and 1994, respectively. In 1986, he joined the Mechanical Engineering Laboratory, Ministry of International Trade and Industry. He is currently scientific group leader of Autonomous Behavior Control Research Group and Co-Director of AIST/IS-CNRS/ST2I Joint French-Japanese Robotics Laboratory (JRL), Intelligent Systems Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), at Tsukuba, Japan. He is also an adjunctive professor of Cooperative Graduate School at University of Tsukuba. From November 1994 to October 1995, he was a Visiting Scholar at Robotics Laboratory, Computer Science Department, Stanford University. His research interests include humanoids, human-centered robotics, and intelligent robot systems.

DOI: 10.1002/cav