2008 8th IEEE-RAS International Conference on Humanoid Robots
December 1 ~ 3, 2008 / Daejeon, Korea



# A Handy Humanoid Robot Navigation by Non-interruptive Switching of Guided Point and Synergetic Points

Tomomichi Sugihara and Hidehito Kobayashi
*School of Information Science and Electrical Engineering, Kyushu University*
*744 Moto'oka, Nishi-ku, Fukuoka, Japan*
zhidao@ieee.org

***Abstract*— A handy humanoid robot navigation scheme is proposed. The whole-body motion of the robot is orchestrated from a motion of a manually guided body part by an input device and that of synergetically coordinated body parts. Even though a simple input device such as a joystick is used, a variety of motions are realized in realtime by switching the directly-guided point without interruptions. The key technique to guarantee the motion achievement is to design a motion segment with a duration from the beginning to the termination from a set of consistent subgoals. An interactive humanoid navigation system with a joystick is presented based on the scheme. The validity of the proposed method is shown through both simulation and real robot operations.**

***Index Terms*— Humanoid robot, motion navigation, online motion planning, navigation interface.**

## I. Introduction

Humanoid robots have potentials to execute high-level tasks as human operators do. A lot of efforts should be continued in order to take them into various scenes of our lives. In the meantime, it is realistic that operators assist robots' brain works including decision making and motion planning rather than they rely on the robot autonomy. An interactive humanoid robot navigation, however, is difficult since they have as complicated structures with a number of joints as humans, in addition to the risks to falling down.

The navigation requires a map from the input variables issued by the operation device to the configuration variables of the target machine. Since humanoid robots have similar form with humans, the problem of dimensionality between the input and output is possibly moderated if the operator's body motion is directly mapped as Fig.1 illustrates. Based on this idea, some humanoid navigation systems have been developed[1][2]. They necessitate large input devices e.g., the motion capture system, to digitize the operator's motion information. It is also a burden for the operators that they have to do the same behavior with what they want the robot to do. Although some small puppet-type input devices have been proposed [3][4], it is also troublesome that the operator needs to handle many body parts of the puppet simultaneously.

On the other hand, several navigation schemes which use handy devices with a small number of input variables e.g., joysticks [5][6][7][8][9] have been developed. Since the number of the input variables is smaller than that of the output variables, a variety of robot motions is rather sacrificed. In order to cover this drawback, Neo et al.[6] focused on the

*This work is supported by "The Kyushu University Research Superstar Program (SSP)", based on the budget of Kyushu University allocated under President's initiative.

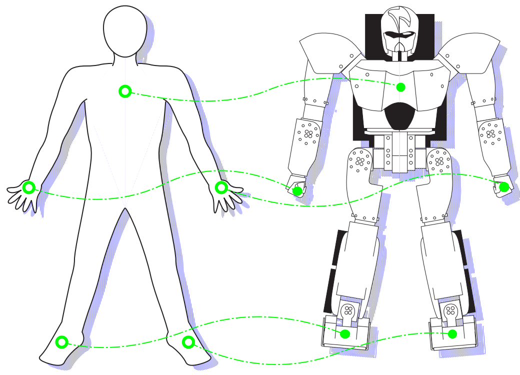



Fig. 1. A humanoid robot control by tracing a navigator's motion requires a large input devices and an overload to the navigator.

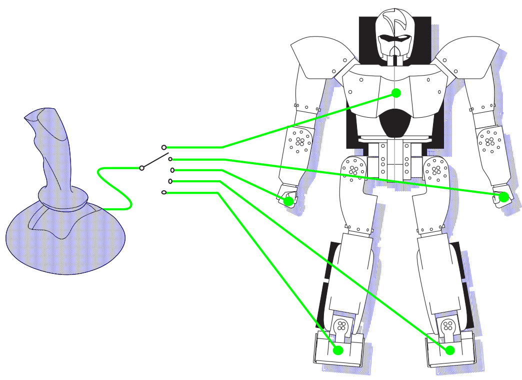


Fig. 2. An input from a device is assigned to a guided point on the robot. By switching the assignment, a variety of motion is orchestrated.

fact that humans pay most attention to a certain body point which is specific to the task to be executed, and move the other parts unconsciously in accordance with equilibratory sense, self-collision avoidance, and so forth. The input are assigned to the body part which the operator concentrates on to guide (let us call it the guided point, hereafter), and the other parts (let us call it the synergetic points, hereafter) are autonomously generated complying with the movement of the guided point as Fig.2 depicts. The operator can navigate the humanoid robot to execute various tasks by switching the guided point. This scheme explains that the operator doesn't necessarily have to care the whole possible robot configurations in ordinary tasks; the key issue is to guide an important body part.

Neo et al.[6] and Yokoi et al.[7] proposed their remote controlling methods of humanoid robots based on this framework. They control the guided point to track the velocity



978-1-4244-2822-9/08/$25.00 ©2008 IEEE

command issued by the input device exactly. The synergetic points automatically move in order to compensate the deviation of the whole-body momentum and to keep sufficient work space. However, they do not guarantee the velocity command to be continuous and consistent with the robot status. A careless command might lead the robot motion into bankruptcy, so that the operator has to monitor the robot status and issue the command deliberately. Switching the guided point needs an interruption of the robot motion. The actual operations are necessarily centered around quasi-static standing motions. When the operation involves step motions, it has to be explicitly commanded. It is hard to facilitate smooth operations on such a system that a big compromise from the operator towards the robot is required due to the above restrictions on motions.

The navigation is basically aimed at the task execution rather than a precise motion tracking in the velocity level. It is preferable that the robot judges if it has to make a step by itself. An important property for the scheme is to design motions of the navigation points with a duration from the beginning to the termination by setting consistent subgoals on site. Based on this idea, this paper proposes a handy humanoid robot navigation method to orchestrate a variety of motions by a simple input device. No matter how the input device is manipulated, the online motion planning is not bankrupted. Operators don't need to interrupt the robot motion to switch the guided point. An interactive humanoid robot navigation system will be also presented.

## II. Humanoid Robot Navigation by Switching the Guided Point

### A. Motion design based on direct-guidance and synergetics

Let us denote the variables from the input device by $\boldsymbol{x}$, and the robot joint angle vector to be actuated by $\boldsymbol{q}$, respectively. Suppose their dimensions are $n_I$ and $n_C$, respectively. The degree-of-freedom of the robot is $n_C + 6$, where the additional 6-DOF means the translational and rotational motion of the trunk. It is thought that $n_I \simeq 6$ and $n_C \simeq 30$ in typical cases. In this paper, a control system of a humanoid robot with 20 motors by a joystick with 6 input variables will be presented, so that $n_I = 6$ and $n_C = 20$. Also, let us represent combinations of the position and attitude of the guided point and the synergetic points by $\boldsymbol{r}_G$ and $\boldsymbol{r}_S$, respectively. Suppose their dimensions are $n_G$ and $n_S$, respectively. We assume that the motion of the guided point can be navigated by the input variable $\boldsymbol{x}$, and that the robot motion can be described by $\boldsymbol{r} \equiv [\boldsymbol{r}_G^{\mathrm{T}}\ \boldsymbol{r}_S^{\mathrm{T}}]^{\mathrm{T}}$. Necessary conditions are $n_G \leq n_I$ and $n_G + n_S \leq n_C + 6$. In particular, $\boldsymbol{q}$ is uniquely determined from $\boldsymbol{r}$ if and only if $n_G + n_S = n_C + 6$; it is an inverse kinematics problem. If $n_G + n_S < n_C + 6$, a redundancy should be resolved by any means. This paper doesn't discuss it in deep.

The fact that the robot has six unactuated coordinates poses a severe dynamical constraint on the motion coordination. As noted in section I, consistent subgoals of the navigation points, namely, the set of the guided point and the synergetic points, should be given in order to build a robust robot navigation scheme which accepts any commands from the input device. Let us discuss this issue through an example

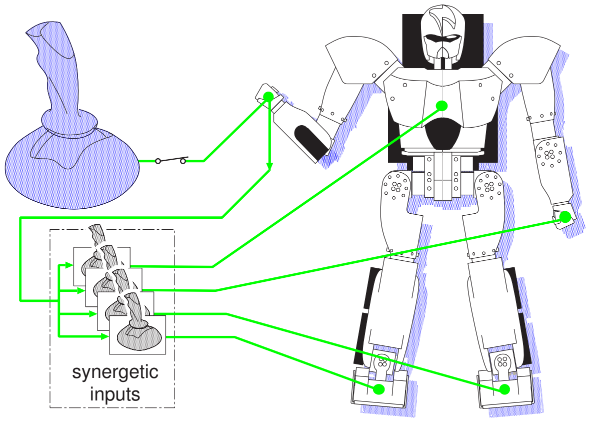



Fig. 3. Proposed humanoid navigation method with guided point switching. Motion of the guided point works as a virtual input to the synergetic points.

motion in which the robot lifts up a foot from a stationarily standing state on both feet. Since the robot mass is evenly loaded on both feet in the initial state, it will tip over by lacking the supporting torque if a foot is carelessly lifted. This should be avoided by kicking the ground to gain an initial velocity of the center of mass (COM) towards the pivoting foot before lifting up the foot. This example suggests an interesting paradox that a high responsivity of the robot to the input rather reduces the motion responsivity since the operator has to command the above complicated sequence manually. In qualitative sense, it means that the motion command should be given with a certain duration so that the robot can autonomously make an anticipation. Based on this idea, let the desired goal position and attitude of the guided point after $T$[s], ${}^d\boldsymbol{r}_G$, be mapped from the input $\boldsymbol{x}$, while the goal position and attitude of synergetic points after $T$[s], ${}^d\boldsymbol{r}_S$ are mapped from ${}^d\boldsymbol{r}_G$ as follows:

$$ {}^d\boldsymbol{r}_G = \boldsymbol{f}_G(\boldsymbol{x}) \tag{1} $$

$$ {}^d\boldsymbol{r}_S = \boldsymbol{f}_S({}^d\boldsymbol{r}_G), \tag{2} $$

where $\boldsymbol{f}_G(\boldsymbol{x})$ and $\boldsymbol{f}_S({}^d\boldsymbol{r}_G)$ represent guiding and synergetic mapping rules, respectively. The guided point's motion works as a virtual navigation input to the synergetic points as Fig.3 depicts. As the result, ${}^d\boldsymbol{r} \equiv [{}^d\boldsymbol{r}_G^{\mathrm{T}}\ {}^d\boldsymbol{r}_S^{\mathrm{T}}]^{\mathrm{T}}$ is determined only from $\boldsymbol{x}$. The motion continuity is guaranteed if a motion trajectory $\boldsymbol{r}(t)$ satisfies the given initial state of the robot, $\boldsymbol{r}(0) = \boldsymbol{r}_0$ and $\dot{\boldsymbol{r}}(0) = \boldsymbol{v}_0$. For simplicity, suppose that the desired goal velocities of any parts are zero. Thus, $\boldsymbol{r}(t)$ should satisfy $\boldsymbol{r}(T) \simeq {}^d\boldsymbol{r}$ and $\dot{\boldsymbol{r}}(T) \simeq \boldsymbol{0}$. When a switching command of the guided point is issued during the motion, it is applied to the next motion segment. The motion trajectory planning method is described in section III-A. A default $T$ is given a priori. If the planned trajectory is physically unacceptable, $T$ is modified according to the method which will be explained in section III-B. The main problem is how to design consistent combinations of the mapping rules $\boldsymbol{f}_G(\boldsymbol{x})$ and $\boldsymbol{f}_S({}^d\boldsymbol{r}_G)$ for possible combinations of the guided point and the synergetic points.

### B. Guiding modes and Synergetic rules

The guiding mode in this paper means the possible choices of the guided part and the synergetic parts, and the navigation rules to decide the consistent goal configurations of those parts. Assuming that the robot works in daily human environments, we choose the candidates of the navigation points for both hands, both feet and the trunk position and attitude. Let us represent the positions and attitudes of the right hand, the left hand, the right foot, the left foot, and the trunk by $\boldsymbol{r}_{LH}$, $\boldsymbol{r}_{RH}$, $\boldsymbol{r}_{LF}$, $\boldsymbol{r}_{RF}$ and $\boldsymbol{r}_{B}$, respectively, where $\boldsymbol{r}_* = [\boldsymbol{p}_*^{\mathrm{T}}\ \boldsymbol{\xi}_*^{\mathrm{T}}]^{\mathrm{T}}$, $\boldsymbol{p}_*$ and $\boldsymbol{\xi}_*$ are a 3-dimensional vector for the position and a 3-dimensional quasi-vector for the attitude (ex. an angle-axis vector) of the part $*$, respectively. Suppose the desired value of $\boldsymbol{r}_*$ is ${}^d\boldsymbol{r}_* = [{}^d\boldsymbol{p}_*^{\mathrm{T}}\ {}^d\boldsymbol{\xi}_*^{\mathrm{T}}]^{\mathrm{T}}$. The guided point is selected from those five parts, while the rest of them are the synergetic points. In case that the left foot is directly guided, for instance, $\boldsymbol{r}_G = \boldsymbol{r}_{LF}$ and $\boldsymbol{r}_S = [\boldsymbol{r}_{LH}^{\mathrm{T}}\ \boldsymbol{r}_{RH}^{\mathrm{T}}\ \boldsymbol{r}_{RF}^{\mathrm{T}}\ \boldsymbol{r}_{B}^{\mathrm{T}}]^{\mathrm{T}}$. Then, we define the foot-guided mode, the trunk-guided mode and the hand-guided mode. The map $\boldsymbol{f}_G(\boldsymbol{x})$ from the input variable to the goal configuration in each mode is designed under the following assumptions.

1) Manipulations of tools and objects are outside the scope.
2) The hopping motions are not considered, i.e. at least one foot is in contact with the ground at any instance.
3) The dual-arm operation is not supposed, i.e. at most only one hand is directly guided.
4) The information about the operation environment is known when deciding the goal configurations.
5) The trunk position is represented by COM.

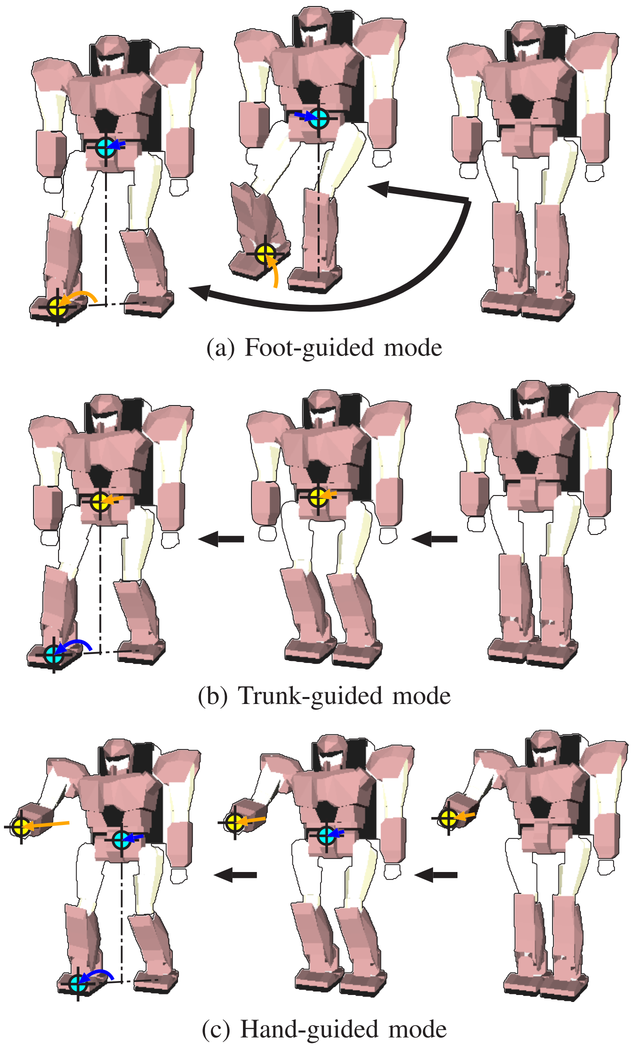
In order to take the achievable range into account and to avoid self-collision and cross-of-extremities, the work space of the navigation points are restricted, but only at the goal configurations for simplicity. The maps $\boldsymbol{f}_S({}^d\boldsymbol{r}_G)$ from the goal of each guided point to that of the corresponding synergetic points are designed as follows.

**Foot-guided mode** is to guide one foot directly. It is useful in cases where the robot should place feet in narrow areas with many obstacles, or on discrete footholds. The opposite foot to the guided foot will be the stance foot which will keep the current configuration on the ground. The guiding command will be bounded by a region which is set in advance with self-collision avoidance and reachability taken into account. If it exceeds the boundary, it will be discarded. As Fig.4(a) shows, the goal position of COM is set above the stance foot in case where the robot will stand on one foot, or above the midpoint of the both feet in case where the guided foot is navigated onto the ground. In the meantime, COM will keep the same height. If the left foot is navigated, for instance:

$$
{}^d\boldsymbol{p}_B = \begin{cases} \boldsymbol{p}_{RF} + \{(\boldsymbol{p}_B - \boldsymbol{p}_{RF})^{\mathrm{T}}\hat{\boldsymbol{g}}\}\hat{\boldsymbol{g}} & \text{(one-foot standing)} \\ \dfrac{{}^d\boldsymbol{p}_{LF} + \boldsymbol{p}_{RF}}{2} + \{(\boldsymbol{p}_B - \boldsymbol{p}_{RF})^{\mathrm{T}}\hat{\boldsymbol{g}}\}\hat{\boldsymbol{g}} & \text{(one-step)} \end{cases} \tag{3}
$$

where $\hat{\boldsymbol{g}}$ is a normalized direction vector of gravity. The goal horizontal orientation of the trunk is set for the middle of both feet, while the goal inclination are set for the current angles. For both hands, the goals are set for the current relative configurations to the trunk. The maximum lift height of the guided foot is also set in advance.

**The trunk-guided mode** is to guide COM position and the trunk attitude. It is useful for transportations and bending



(a) Foot-guided mode

(b) Trunk-guided mode

(c) Hand-guided mode

Fig. 4. Guiding modes and subgoals of navigation points.

motions in such environments that the operator doesn't need to care the leg room. If the goal COM position is set outside the stand-on region, which is defined as a shrunk supporting region as illustrated in Fig.5, the farther foot from the goal COM position will step out. As Fig.4(b) shows, both feet keep the current grounding configurations if a stepping motion is not required. Otherwise, the goal position of the stepping foot is set in such a manner that the midpoint of both feet is located below the goal COM position. If the left foot steps, for instance, its goal position is defined as follows:

$$
{}^d\boldsymbol{p}_{LF} = 2[{}^d\boldsymbol{p}_B - \{(\boldsymbol{p}_B - \boldsymbol{p}_{RF})^{\mathrm{T}}\hat{\boldsymbol{g}}\}\hat{\boldsymbol{g}}] - \boldsymbol{p}_{RF}. \tag{4}
$$

It is discarded with the original COM command if it exceeds the boundary. For both hands, the goals are set for the home positions and attitudes, which are fixed in the body frame.

**Hand-guided mode** is to guide a hand. COM is automatically attracted by the commanded goal position of the hand in the following way. Suppose the left foot is directly guided. A distance $l$ from the current left shoulder position $\boldsymbol{p}_{LS}$ to the left hand is computed as:

$$
l = ||{}^d\boldsymbol{p}_{LH} - \boldsymbol{p}_{LS}||. \tag{5}
$$

The goal position of the left shoulder is defined in accordance with the maximum acceptable distance between the

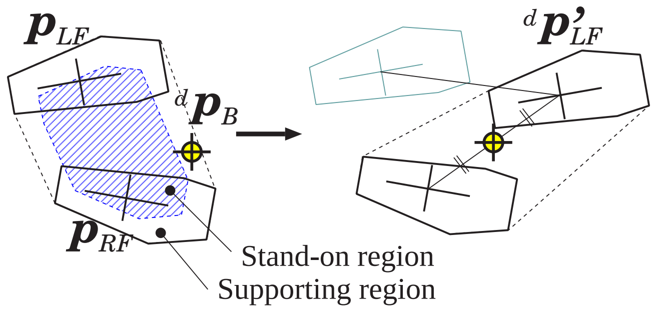



Fig. 5. Autonomous foot stepping in the trunk-guided mode.

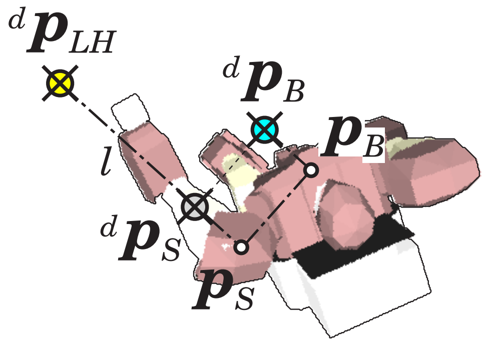



Fig. 6. Autonomous COM traction in hand-guided mode.

hand and the shoulder $l_{max}$ given in advance as follows:

$$^d p_{LS} = \begin{cases} p_{LS} & (l \le l_{max}) \\ p_{LS} + \dfrac{l - l_{max}}{l}\left(^d p_{LH} - p_{LS}\right) & (l > l_{max}) \end{cases} . \quad (6)$$

The goal COM position is defined from $^d p_{LS}$ as follows:

$$^d p_B = p_B + \left(^d p_{LS} - p_{LS}\right) . \quad (7)$$

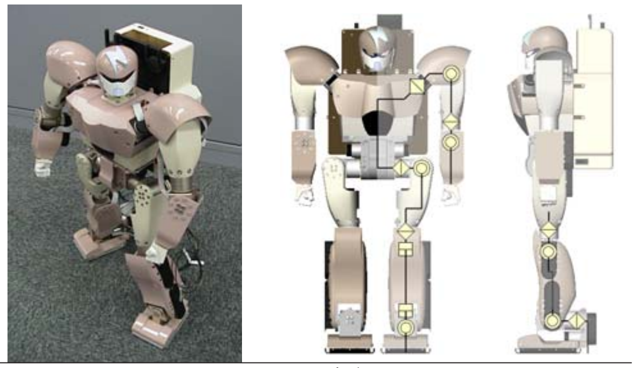
Fig.4(c) depicts the above procedure. Whether a step is needed and whether the command should be discarded are also judged from $^d p_B$ in the same way with that in the trunk-guided mode. The opposite hand keeps the current relative configuration to the trunk.

## III. Online Motion Trajectory Planning

### A. A stepwise three-dimensional trajectory design[10]

Given the goal set of the navigation points $^d r$ by the previous rules, the whole-body motion trajectory which smoothly connects the initial robot state and the goal state in time $T$ is planned immediately after the goal is defined. Since the both hands and feet are related to the stance condition and the task achievement, they are designed by quintic functions or cycloid curves which satisfy the given boundary conditions, for instance.

The dynamical consistency of the robot motion is approximately taken into account as the problem of the COM trajectory planning. We use the method by Terada et al.[10]. It can achieve a stepwise biped motion planning, which is in general the boundary value problem of a differential equation under an inequality constraint. Two fundamental techniques are the dynamical 3D-symmetrization proposed by Terada and Kuniyoshi[11], and the boundary condition relaxation method by Sugihara and Nakamura[12]. The former is to obtain a general solution of the differential equation which can represent a three-dimensional motion of COM in accordance with the spatial symmetry of the homogeneous solution. The latter is to solve the boundary value problem under an inequality constraint by relaxing the boundary condition, focusing on the degree of severities of each condition. As the result, trajectories of COM, ZMP and extremities are simultaneously computed in addition to the supporting region. Refer to each paper for the details.



| | |
|---|---|
| Name: | mighty |
| Height: | 580 [mm] |
| Weight: | 6.5 [kg] |
| Number of actuated joints: | 20 (8 for arms, 12 for legs) |

Fig. 7. External view and main specifications of the robot

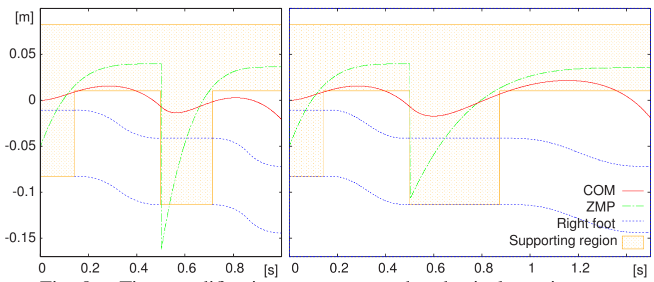



Fig. 8. Time modification to guarantee the physical consistency.

### B. Auto-correction of duration for dynamical consistency

Although the motion planning method in the previous subsection compromises the given boundary condition and the inequality constraint, it doesn't necessarily guarantee the constraint to be satisfied. In particular, the computed ZMP trajectory possibly passes outside of the planned supporting region. Such a motion is not physically acceptable. This subsection presents a heuristic technique to resolve it by modifying the motion duration $T$.

It is known that the horizontal distance between COM and ZMP is approximately proportional to the COM acceleration [13]. If the planned ZMP is unacceptably far from COM, it means that the required horizontal acceleration exceeds the physical limit. Then, it is expected that the magnitude of COM acceleration is suppressed if the motion duration $T$ for the same COM transportation is prolonged. This idea is implemented as follows.

1) If the planned ZMP trajectory lies within the planned supporting region, terminate.
2) If $T \ge T_{max}$, where $T_{max}$ is the pre-defined maximum limitation of $T$, abort.

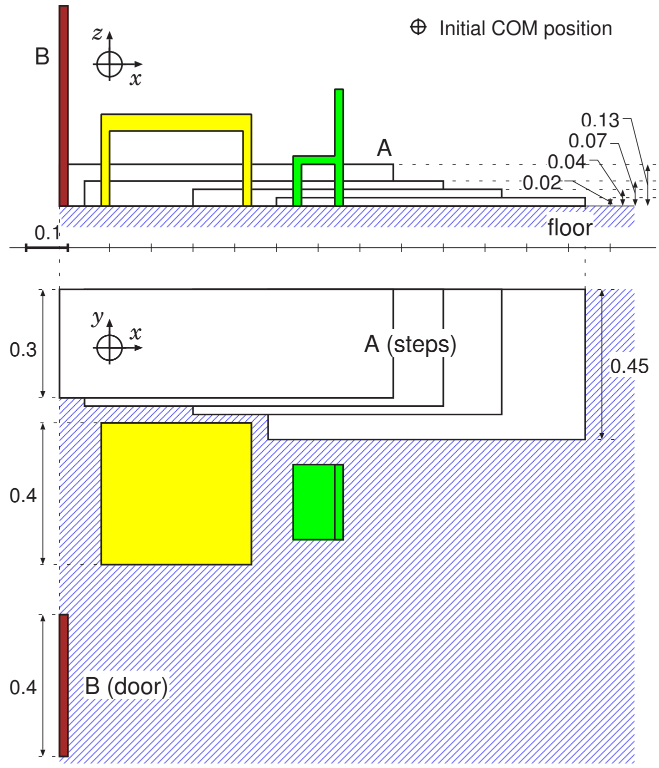



Fig. 9. Description of the experimental environment model.

3) $T \leftarrow T + \Delta T$. Then, return to 1).

where $\Delta T$ is defined in advance. In order to avoid the abnormal case where $T \geq T_{max}$, the boundary of the goal commands against excess inputs should be carefully set.

## IV. Implementation of Navigation System

This section describes a humanoid robot navigation system which we have implemented in order to verify the proposed method. We adopted a commercial joystick Aviator AV8R made by Saitek for the input device. It is equipped with a main stick which can input heading-pitch-bank angles, a miniature sub stick attached at the tip of the main stick to input up-down and sideway movements, two throttles, one toggle switch, four 3-value momentary switches, and three trigger buttons. Our system uses the sub stick for the horizontal position command, one of the momentary switches for the vertical position command, the main stick for the attitude command, the throttles for the foot-lifting height commands, and the toggle switch for the navigation mode switching, respectively. The joystick sends the device status to PC via USB.

The trajectory planning software reads the current joystick status and plans the next motion segment as soon as the previous motion terminates. A default value is set for the motion segment term $T$, which is modified based on the method in section III-B if necessary. The trajectory is quantized by a sampling time $\Delta t$[s], and the inverse kinematics is solved for the desired configurations of the navigation points at each time step by COM Jacobian matrix[14] and Yamane and Nakamura's method[15]. The default $T$ and $\Delta t$ were set for 0.5[s] and 0.005[s], respectively, conforming to the actual robot system. When we implemented the software on a PC with Athlon 64 X2 3800+ and 2GB RAM, running on Linux, the trajectory planning for each motion segment took about 0.0003[s]. Though it inserts blanks between the motion segments, it is thought to be short enough to be neglected.

For the humanoid robot, we used mighty[16], which Fig.7 shows. The trajectory planning sends the whole desired joint angles to the control system of mighty via the socket communication. Since mighty has only four degrees of freedom in each arm, the commands about hand orientations are ignored.

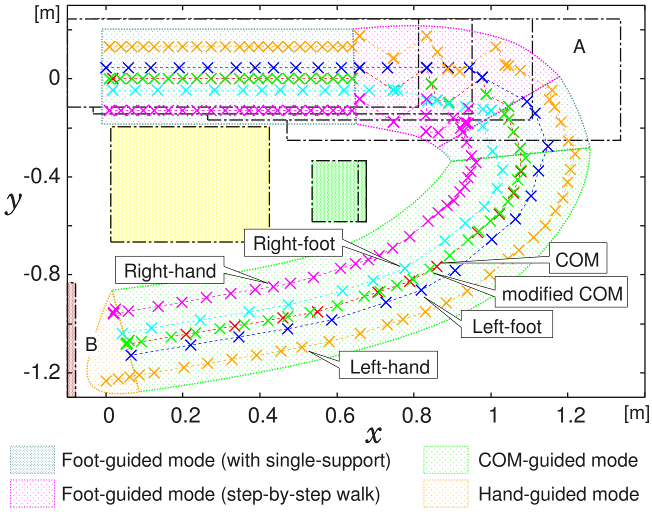



Fig. 10. Loci of COM, feet and hands planned by navigation system.

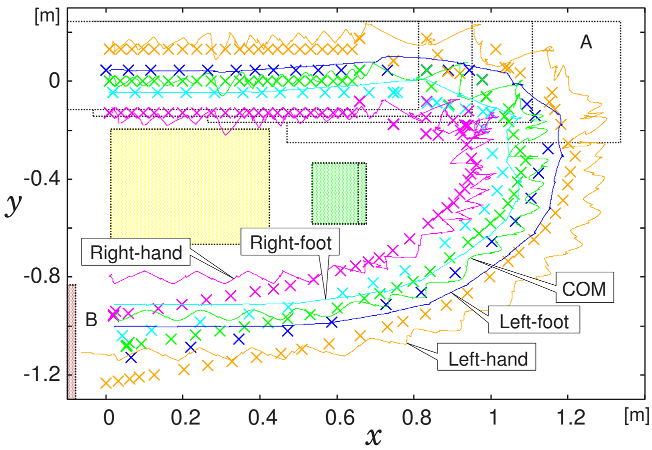



Fig. 11. Loci of COM, feet and hands in the simulation.

## V. Simulations and Experiments

Some simulations and experiments have been conducted on our navigation system introduced in the previous section. The system can substitute the robot for OpenHRP[17] as a communication target of the navigation software, so that a seamless simulation with the actual system is available.

We firstly verified if the motion term modification method in section III-B can deform the desired ZMP trajectory to be within the planned supporting region. The sample test motion is that the robot steps out the right foot 0.03[m] rightward, and then another 0.03[m] rightward. The left side of Fig.8 shows the loci of COM and ZMP in case where the motion term was not modified. The ZMP trajectory passes outside of the planned supporting region in the beginning of the second step; it is hard to forecast the actual robot motion if the trajectory is applied. The result of the motion term

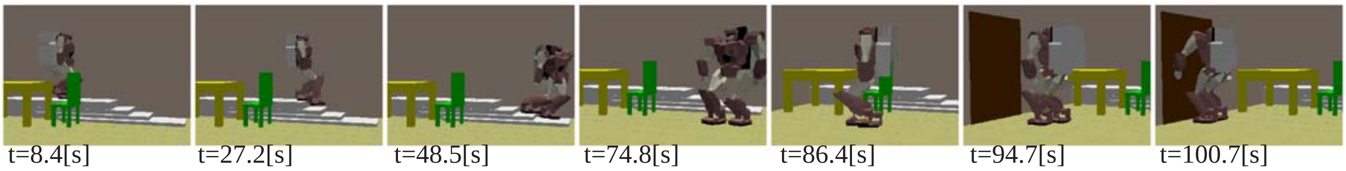



Fig. 12. Snapshots of a simulation of transportation-reaching task execution.

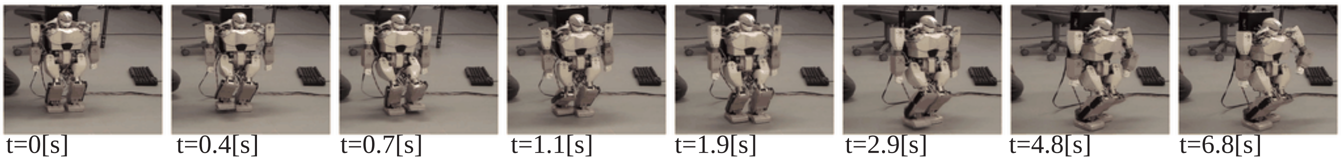



Fig. 13. Snapshots of an experiment of transportation-reaching task execution.

modification can be seen in the right side of the same figure, where the ZMP exists in the planned supporting region at every time. In this case, the motion term of the second step was changed to 1.0[s].

Then, a task operation including foot steps and a reaching in an environment illustrated by Fig.9 was simulated. The scenario is that the robot goes down the footsteps A, walks across the room and reaches the door B by the left hand. The sequential subgoals are shown in Fig.10. In order to go down A, the foot-guided mode was applied. In the beginning, foot placements were navigated step by step on the flat stage, and then, each foot was carefully guided onto the stairs by utilizing single-supports. After that, the robot body was guided toward the door B across the room by the trunk-guided mode. Both feet automatically stepped to carry COM. The subgoals of COM were also auto-corrected with the foot placement taken into account. For the last, the left hand was guided to approach the door B. The loci of COM, feet and hands of the simulation are plotted in Fig.11. Their deviations from the planned trajectories are mainly due to the slippage neglected in the planning phase. Note that the simulation and navigation were simultaneously conducted. The navigation trajectory were deviated from the target B, since the stepwise commands were accumulated. However, the commands were concatenated to the actual robot state, so that the robot was able to reach B. Fig.12 shows snapshots of the simulated robot motion.

Finally, a similar task was operated with the real robot on a flat ground. Fig.13 shows snapshots of the experiment. The robot achieved the operation without falling down.

## VI. Conclusion

This paper proposed a humanoid robot navigation scheme, which facilitates rich humanoid behaviors for various task executions even by a handy input device like a joystick. Operators can switch a body part which is directly guided by the input device during operations without interruption, while the other body parts engage to the guided-part in accordance with synergetic rules. A key issue for a robust online motion planning which can accept any commands at any timing from the input device is to design motions of the navigation points with a duration from the beginning to the termination. Thanks to it, the robot can autonomously judge if it should step or not. Based on the proposed scheme, an interactive humanoid robot navigation system was also developed.

## References

[1] Tachi et al., "Development of Telexistence Cockpit for Humanoid Robot Control," in *Proceedings of the 32nd International Symposium on Robotics*, 2001, pp. 1483–1488.

[2] Kurihara et al., "Optical Cockpit for Humanoid Tele-Operation with Realtime Motion Capture System," in *Proceedings of the JSME Conference on Robotics and Mechatronics*, 2002.

[3] Kagami et al., "AutoBalancer: An Online Dynamic Balance Compensation Scheme for Humanoid Robots," in *Proceedings of the 4th International Workshop on Algorithmic Foundation on Robotics(WAFR'00)*, 2000.

[4] Takubo et al., "Marionette System for Operating and Displaying Robot Whole-Body Motion – Development of Similar Humanoid-Type Device –," in *Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004, pp. 509–514.

[5] Nishiwaki et al., "Online Mixture and Connection of Basic Motions for Humanoid Walking Control by Footprint Specification," in *Proceedings of the 2001 IEEE International Conference on Robotics & Automation*, 2001, pp. 4110–4115.

[6] Neo et al., "Whole-Body Motion Generation Integrating Operator's Intention and Robot's Autonomy in Controlling Humanoid Robots," *IEEE Transaction on Robotics*, vol. 23, no. 4, pp. 763–775, 2007.

[7] Yokoi et al., "A Tele-operated Humanoid Operator," *International Journal of Robotics Research*, vol. 25, no. 5–6, pp. 593–602, 2006.

[8] K. Kai, K. Yamane, and Y. Nakamura, "Choreography of Humanoid Robots with Hand Motion Capture System," in *Proceedings of the JSME Conference on Robotics and Mechatronics*, 2006.

[9] Chestnutt et al., "An Intelligent Joystick for Biped Control," in *Proceedings of the 2006 IEEE International Conference on Robotics & Automation*, 2006, pp. 860–865.

[10] K. Terada, T. Sugihara, and Y. Kuniyoshi, "Online Gait Planning with Boundary Condition Relaxation and Dynamical 3D-Symmetrization," in *25th Annual Conference of Robotics Society of Japan*, 2007.

[11] K. Terada and Y. Kuniyoshi, "Online Gait Planning with Dynamical 3D-Symmetrization Method," in *Proceedings of 2007 IEEE-RAS International Conference on Humanoid Robots*, 2007.

[12] T. Sugihara and Y. Nakamura, "A Fast Online Gait Planning with Boundary Condition Relaxation for Humanoid Robots," in *Proceedings of the 2005 IEEE International Conference on Robotics & Automation*, 2005, pp. 306–311.

[13] K. Mitobe, G. Capi, and Y. Nasu, "Control of Walking Robots based on Manipulation of the Zero Moment Point," *Robotica*, vol. 18, pp. 651–657, 2000.

[14] R. Boulic, R. Mas, and D. Thalmann, "Inverse Kinetics for Center of Mass Position Control and Posture Optimization," in *Proceedings of European Workshop on Combined Real and Synthetic Image Processing for Broadcast and Video Production*, November 1994.

[15] K. Yamane and Y. Nakamura, "Natural Motion Animation through Constraining and Deconstraining at will," *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, no. 3, pp. 352–360, 2003.

[16] T. Sugihara, K. Yamamoto, and Y. Nakamura, "Hardware design of high performance miniature anthropomorphic robots," *Robotics and Autonomous System*, vol. 56, pp. 82–94, 2007.

[17] F. Kanehiro, H. Hirukawa, and S. Kajita, "OpenHRP: Open Architecture Humanoid Robotics Platform," *The International Journal of Robotics Research*, vol. 23, no. 2, pp. 155–165, 2004.