Real-time Footstep Planning and Following for Navigation of Humanoid Robots

Young-Dae Hong[†]

Abstract – This paper proposes novel real-time footstep planning and following methods for the navigation of humanoid robots. A footstep command is defined by a walking direction and step lengths for footstep planning. The walking direction is determined by a uni-vector field navigation method, and the allowable yawing range caused by hardware limitation is considered. The lateral step length is determined to avoid collisions between the two legs while walking. The sagittal step length is modified by a binary search algorithm when collision occurs between the robot body and obstacles in a narrow space. If the robot body still collides with obstacles despite the modification of the sagittal step length, the lateral step length is shifted at the next footstep. For footstep following, a walking pattern generator based on a 3-D linear inverted pendulum model is utilized, which can generate modifiable walking patterns using the zero-moment point variation scheme. Therefore, it enables a humanoid robot to follow the footstep command planned for each footstep. The effectiveness of the proposed method is verified through simulation and experiment.

Keywords: Humanoid robot, Footstep planning, Uni-vector field navigation method, Binary search algorithm, Footstep following, Modifiable walking pattern

1. Introduction

For navigation of humanoid robots in real human environments, footstep planning in such environment with obstacles is one of the most important issues in the field of humanoid robot research. Attempts to use sets of predetermined step lengths and walking patterns have been introduced [1, 2]. Using A* search algorithm, several methods have computed footstep sequences in which feasible regions of foot positions were predefined, and a few foot positions were then selected as a discrete set [3, 4]. In addition, by using a rapidly exploring random tree algorithm, randomized sampling-based methods have solved the complexity problem of the A* search algorithm for footstep planning [5-7]. However, these previous methods must precalculate the discrete set of feasible footstep positions and corresponding leg-joint trajectories for footstep transition. In addition, they did not consider the elements for footstep following, such as the hardware limitation of the robot, collisions between the two legs, and collisions between the robot body and obstacles while walking. Meanwhile, an evolutionary optimization algorithm has been employed to plan the optimal footstep sequences [8, 9]. However, real-time footstep planning was impossible because of the high computing power requirement of evolutionary optimization.

Therefore, the present paper proposes novel real-time footstep planning and following methods for the navigation

of humanoid robots. A footstep command is defined by a walking direction and step lengths for footstep planning. The walking direction is determined by a uni-vector field navigation method [10], and the allowable yawing range due to hardware limitation is considered. The lateral step length is determined to avoid collision between the two legs while walking by maintaining the same distance between the left and right feet. The sagittal step length is predefined, and it is modified by a binary search algorithm when collision between the robot body and obstacles occurs in a narrow space. If the robot body still collides with obstacles despite the modification of the sagittal step length, the lateral step length is shifted at the next footstep. For footstep following, a walking pattern generator based on a 3-D linear inverted pendulum model (LIPM) [11] is utilized, which can generate modifiable walking patterns using the zero-moment point (ZMP) variation scheme [12-14]. Therefore, it enables a humanoid robot to follow the footstep command planned for each footstep.

This paper is organized as follows. Section 2 describes the proposed footstep planning method. Section 3 presents the footstep following method, and Section 4 explains the overall procedure of the proposed method. Section 5 presents the verification of the effectiveness of the proposed method through simulation and experiment. Finally, Section 6 provides the conclusion.

2. Footstep Planning

For the footstep planning, walking direction $\theta_{r/l}$,

[†] Corresponding Author: Dept. of Electrical and Computer Engineering, Ajou University, Korea. (ydhong@ajou.ac.kr)
Received: March 16, 2015; Accepted: May 21, 2015

sagittal step length $S_{r/l}$, and lateral step length $L_{r/l}$ of the right/left swing leg are defined as a footstep command, which is expressed as follows:

$$(\theta_{r/l}, S_{r/l}, L_{r/l}). \quad (1)$$

The footstep command for each footstep is generated by the proposed footstep planning method.

2.1 Determination of walking direction

A uni-vector field navigation method is adopted to determine the walking direction. It is appropriate for execution in real-time owing to its low computing power [10]. Fig. 1 shows a move-to-goal uni-vector field (MUF) that enables the robot to arrive at the goal and an avoid-obstacle uni-vector field (AUF) that makes the robot avoid obstacles. The MUF and AUF are generated by an attractive uni-vector field and a hyperbolic spiral uni-vector field, respectively, as follows:

$$\mathbf{u}_{muf}(\mathbf{p}) = [-\cos\theta_m(\mathbf{p}) \quad -\sin\theta_m(\mathbf{p})]^T \quad (2)$$

$$\mathbf{u}_{auf}(\mathbf{p}) = \begin{cases} [-\cos\varphi_h(\mathbf{p}) \quad -\sin\varphi_h(\mathbf{p})]^T, & \text{if } d_o < \rho_{op} \leq d_o + d_e + d_b \\ & \text{and } \rho_{pg} + d_o \geq \rho_{og} \\ [0 \ 0]^T, & \text{otherwise} \end{cases} \quad (3)$$

(a)

(b)

Fig. 1. Uni-vector fields: (a) MUF; (b) AUF.

with

$$\varphi_h(\mathbf{p}) = \begin{cases} \theta_a(\mathbf{p}) \pm \frac{\pi}{2} \left(2 - \frac{d_o + d_e + K_r}{\rho_{op} + K_r} \right), & \text{if } d_o + d_e \leq \rho_{op} \\ \theta_a(\mathbf{p}) \pm \frac{\pi}{2} \left(\frac{\rho_{op} - d_o}{d_e} \right), & \text{if } d_o \leq \rho_{op} \leq d_o + d_e \end{cases}$$

where $\theta_m(\mathbf{p})$ and $\theta_a(\mathbf{p})$ denote the angles from the x -axis of goal $\mathbf{g}$ and the obstacle center $\mathbf{o}$ to position $\mathbf{p}$, respectively; $\varphi_h(\mathbf{p})$ denotes the hyperbolic spiral uni-vector field; d_o , d_e , and d_b are the radius of the obstacle, the predefined radius that determines the size of the spiral, and the size of the boundary of the AUF, respectively; ρ_{ij} denotes the distance between positions $\mathbf{i}$ and $\mathbf{j}$. The $+$ and $-$ signs are determined according to the robot position relative to the obstacle and goal, where, $-$ indicates a clockwise motion and $+$ indicates a counterclockwise motion. Variable K_r determines the smoothness of the contour of the spiral.

Base position $\mathbf{p}_B$ is defined as the midpoint between the left and right feet, and the walking direction at each footstep is then determined by the direction of the total uni-vector field at base position $\mathbf{u}_{uf}(\mathbf{p}_B)$ as follows:

$$\mathbf{u}_{uf}(\mathbf{p}) = \frac{\bar{\mathbf{u}}_{uf}(\mathbf{p})}{|\mathbf{u}_{uf}(\mathbf{p})|} \quad (4)$$

with

$$\bar{\mathbf{u}}_{uf}(\mathbf{p}) = \mathbf{u}_{muf}(\mathbf{p}) + \sum_{i=1}^n \mathbf{u}_{auf_i}(\mathbf{p})$$

where n is the total number of obstacles.

The walking direction determined by the uni-vector field navigation method can exceed the allowable yawing range, which is restricted according to the hardware limitations of the robot. Thus, for footstep following in this case, the walking direction should be modified as follows:

$$\theta_r = \begin{cases} \theta_{\min}, & \text{if } \theta_r > \theta_{\min} \\ -\theta_{\max}, & \text{if } \theta_r < -\theta_{\max} \end{cases}, \quad \theta_l = \begin{cases} \theta_{\max}, & \text{if } \theta_l > \theta_{\max} \\ -\theta_{\min}, & \text{if } \theta_l < -\theta_{\min} \end{cases} \quad (5)$$

where $\theta_{\max}$ and $\theta_{\min}$ denote the maximum and minimum allowable yawing ranges, respectively.

2.2 Determination of step lengths

Collisions between the two legs can occur while the walking direction is changed during walking. Therefore, to solve this problem, the distance between the left and right feet should be maintained at the same value by determining the lateral step length of each footstep, as shown in Fig. 2, which is calculated as follows:

Fig. 2. Modification of the lateral step length to maintain the same distance between the left and right feet.

$$\begin{aligned} L_r &= \begin{cases} -(f_f \sin|\theta_r| + f_i \cos|\theta_r|) + f_i, & \text{if } \theta_r \geq 0 \\ -(f_b \sin|\theta_r| + f_i \cos|\theta_r|) + f_i, & \text{if } \theta_r < 0 \end{cases} \\ L_l &= \begin{cases} f_b \sin|\theta_l| + f_i \cos|\theta_l| - f_i, & \text{if } \theta_l \geq 0 \\ f_f \sin|\theta_l| + f_i \cos|\theta_l| - f_i, & \text{if } \theta_l < 0 \end{cases} \end{aligned} \quad (6)$$

where f_f/f_b and f_o/f_i denote the front/back foot length and outer/inner foot width from the center of the foot, respectively.

Basically, if the robot follows the footsteps planned by the uni-vector field navigation method, obstacle collisions do not occur because the AUF prevents collision with obstacles. However, because the footsteps of the humanoid robot are discrete, the robot can still collide with obstacles in a narrow space. To solve this problem, the sagittal step length $S_{r/l}$ at each footstep is determined by a binary search algorithm, as shown in Algorithm 1, where N is the number of iterations. The binary search algorithm is sufficiently fast for execution in real-time because it is intuitive and simple. The robot body is simplified as a cylinder with radius d_r . The sagittal step length is initially predetermined as normal step length S_n . If the robot body collides with the obstacle, the sagittal step length is modified until collision does not occur. When the robot body still collides with the obstacle in spite of the modification of the sagittal step length, the lateral step length at the next footstep is shifted as a constant value.

3. Footstep Following

To make a humanoid robot follow the footstep command

Fig. 3. 3-D LIPM.

Algorithm 1 Binary search algorithm

```

/* initialization */
 $S_0 \leftarrow 0$ ;
 $S_1 \leftarrow S_n$ ;
 $n \leftarrow 0$ ;
/* start binary search algorithm */
while  $n < N$  do
     $S_m \leftarrow (S_0 + S_1) / 2$ ;
     $S_{r/l} \leftarrow S_m$ ;
    if  $\rho_{op} > d_o + d_r$  then
         $S_0 \leftarrow S_m$ ;
    else if  $\rho_{op} < d_o + d_r$  then
         $S_1 \leftarrow S_m$ ;
    else
        break;
    end if
     $n \leftarrow n + 1$ ;
end while
    
```

planned by the proposed footstep planning method, a walking pattern generator based on the 3-D LIPM is utilized, as shown in Fig. 3 [12-14]. The center-of-mass (COM) motion equations of the 3-D LIPM is given by the ZMP functions $p(t)$ and $q(t)$ as follows:

$$\begin{aligned} \begin{bmatrix} x_f \\ T_c v_f \end{bmatrix} &= \begin{bmatrix} C(t) & S(t) \\ S(t) & C(t) \end{bmatrix} \begin{bmatrix} x_i \\ T_c v_i \end{bmatrix} - \frac{1}{T_c} \begin{bmatrix} S(t) * p(t) \\ C(t) * p(t) \end{bmatrix} \\ \begin{bmatrix} y_f \\ T_c w_f \end{bmatrix} &= \begin{bmatrix} C(t) & S(t) \\ S(t) & C(t) \end{bmatrix} \begin{bmatrix} y_i \\ T_c w_i \end{bmatrix} - \frac{1}{T_c} \begin{bmatrix} S(t) * q(t) \\ C(t) * q(t) \end{bmatrix} \end{aligned} \quad (7)$$

where $(x_i, v_i)/(x_f, v_f)$ and $(y_i, w_i)/(y_f, w_f)$ denote the initial/final COM positions and velocities in the sagittal and lateral planes, respectively. $C(t)$ and $S(t)$ denote $\sinh(t/T_c)$ and $\cosh(t/T_c)$, respectively, where $T_c = \sqrt{Z_c/g}$. * denotes the convolution operator. To follow the footstep commands in which the sagittal and lateral step lengths and the walking direction are varied at every footstep, the walking pattern generator should generate modifiable walking patterns without any additional footsteps to adjust the COM motion in real-time. At the right-hand side of (7), the first and second terms represent the homogeneous solution component and the particular solution component of the dynamic equation of the 3-D LIPM, respectively. In this walking pattern generator, the particular solution component is utilized for the real-time ZMP variation. It allows the ZMP to be varied over the convex hull of the bounded foot region by the ZMP functions $p(t)$ and $q(t)$. From the ZMP variation, the COM position and velocity can be changed throughout a single support motion. Consequently, generating modifiable walking patterns becomes possible. Note that the single and double support times for the footstep following are properly assigned as constant values. It is possible because the walking pattern generator is able to generate the modifiable walking

patterns in which the sagittal and lateral step lengths, and the walking direction are independently changed while maintaining the same walking period at each footstep.

The footstep command is entered into the walking pattern generator at every sampling time. Then, the desired COM position and velocity of the 3-D LIPM for the footstep command are derived [12-14]. Because walking of the humanoid robot is a periodic motion, it can be described by identifying certain particular COM position and velocity of the 3-D LIPM. At the end of each single support phase, the COM position and velocity are identical. Thus, they are regarded as the desired COM position and velocity of the 3-D LIPM. It is assumed that the COM position and velocity are in a steady state and that the ZMP does not vary during the conversion of the footstep command into the desired COM position and velocity of the 3-D LIPM. For the conversion of the footstep command into the desired COM position and velocity, one period of the walking configuration is considered in a steady state. The walking configuration can be fully described by the initial or final COM position and velocity of the 3-D LIPM for both the left and right support phases because the COM position and velocity is in a steady state with no ZMP variation. Next, the COM trajectory that satisfies the desired COM motion is generated by the COM motion equations, and every leg-joint trajectory is provided by inverse kinematics. As a result, the walking pattern generator generates the walking patterns that satisfy the footstep command.

4. Overall Procedure

The overall procedure of the proposed footstep planning and following methods is summarized as shown in Fig. 4, where P denotes the pelvis length of the humanoid robot. ρ_{pBg} and d_g denote the distance between the base position and goal, and the radius of the goal, respectively. For footstep planning, parameters d_e , d_b , and K_r in the AUF are set, and the base position is initialized to $(S_n, P/2)$. At each footstep, sagittal step length $S_{r/l}$ is set as normal step length S_n , and lateral step length $L_{r/l}$ is set to zero. Walking direction $\theta_{r/l}$ is determined by the direction of $\mathbf{u}_{uf}(\mathbf{p}_B)$ and modified by (5) when it exceeds the allowable yawing range. Lateral step length $L_{r/l}$ is determined by (6) for the same distance between the left and right feet. Sagittal step length $S_{r/l}$ is changed by the binary search algorithm when collision occurs between the robot body and the obstacle until the robot body does not collide with the obstacle anymore. If collision still occurs, the lateral step length at the next footstep $L_{l/r}$ is shifted as constant step length L_s . For the footstep following, the footstep command determined by the proposed footstep planning method at each footstep is entered into the walking pattern generator, which generates the leg-joint trajectories for the footstep command. Then, the robot walks with the constant walking

Fig. 4. Flowchart of the overall procedure.

period following the planned footsteps. If the robot arrives at the goal, the sagittal and lateral step lengths are set to zero, and walking is terminated. Consequently, the proposed method can plan the footsteps in real-time and simultaneously make the robot follow the planned footsteps without precalculating the discrete set of feasible footstep positions and corresponding trajectories of every leg-joint for the footstep transition.

5. Simulation and Experiment

The effectiveness of the proposed method is verified through simulation and experiment using the small-sized humanoid robot platform DARwIn-OP, as shown in Fig. 5(a). Its height and weight were 45.5 cm and 2.8 kg, respectively. It has twenty degrees of freedom (DOFs) with

Fig. 5. (a) DARwIn-OP; (b) Simulation model.

MX-28 actuators (two DOFs for head, six DOFs for arms, and twelve DOFs for legs). As the main controller, an Intel Atom Z530 central processing unit, running Linux Ubuntu, is utilized. It communicates with the sub controller, ARM 32-bit Cortex-M3 through universal serial bus connections. The simulated model of DARwIn-OP was produced using Webots, as shown in Fig. 5(b), which is a 3-D robotic simulation software that enables users to perform physical and dynamical simulations [15]. Tables 1 and 2 list the summary of the parameters for footstep planning and those of DARwIn-OP, respectively. The single and double support times were set as 0.4 sec and 0.2 sec, respectively.

Fig. 6 shows the footsteps planned by the proposed method in two environments. In Fig. 6(a), the footsteps were planned in an environment with cylindrical obstacles. In Fig. 6(b), rectangular obstacles were made up of several cylindrical obstacles, and the footsteps were planned in an environment with a narrow passage and a local loop. The figures show that the planned footsteps made the robot arrive at the goal without obstacle collision. The walking direction at each footstep was determined by the value within the allowable yawing range, and the lateral step

Table 1. Parameters for footstep planning

d_c	d_b	K_r	S_n [cm]	L_s [cm]	θ_{max} [°]	θ_{min} [°]
10.0	30.0	30.0	4.0	3.0	15.0	5.0

Table 2. Parameters of DARwIn-OP

P [cm]	f_f [cm]	f_b [cm]	f_o [cm]	f_i [cm]	d_r [cm]
7.4	5.2	5.2	4.4	2.2	15.0 (margin: 2.0)

Fig. 6. Planned footsteps in environments with (a) cylindrical obstacles and (b) a narrow passage and a local loop.

Fig. 7. Snapshots of the walking simulation results in environments with (a) cylindrical obstacles and (b) a narrow passage and a local loop. (left to right and top to bottom).

length was determined to maintain the same distance between the left and right feet. In addition, for collision avoidance between the robot body and obstacles in a narrow space, the sagittal and lateral step lengths were varied. In Fig. 6(a), the sagittal step length was changed six times to 1.9, 2.1, 0.0, 3.0, 0.0, and 0.0 cm at the 3rd, 4th, 5th, 41st, 42nd, and 43rd footsteps, respectively, by the binary search algorithm to avoid collision between the robot body and the obstacles. We note that the thick circles depict the robot body at the footstep when the sagittal step length was changed. The robot body still collided with the obstacle at the 42nd footstep in spite of the change in the sagittal step length to 0.0 cm. Accordingly, the lateral step length at the 43rd footstep was shifted to 3.0 cm. Consequently, the robot avoided the obstacle in the narrow space, and it was then able to arrive at the goal.

Fig. 7 shows snapshots of the walking simulation results in the two environments. The humanoid robot successfully followed each footstep planned by the proposed method in real-time. The walking directions were not restricted by the hardware limitations of the robot because the footstep was planned considering the allowable yawing range. Collision between the two legs did not occur while walking, and the robot body did not collide with the obstacles by adjusting the step lengths. Fig. 8 shows snapshots of the walking experiment result in the environment with cylindrical obstacles. Small position errors existed because of the slip between the sole of the robot and the floor surface

Fig. 8. Snapshots of the walking experiment result in the environment with cylindrical obstacles. (left to right and top to bottom).

compared with the walking simulation result shown in Fig. 7(a). However, the humanoid robot successfully arrived at the goal without obstacle collisions following the planned footsteps in real-time.

6. Conclusion

This paper has proposed a method for planning the footsteps of humanoid robots and following them in real-time. The uni-vector field navigation method was utilized to determine the walking direction, and the walking direction was varied according to the allowable yawing range. The lateral step length was determined to maintain the same distance between the left and right feet for collision avoidance between the two legs while walking. The sagittal step length was determined by the binary search algorithm, and the lateral step length was shifted to avoid collision between the robot body and the obstacles in a narrow space. The walking pattern generator based on the 3-D LIPM generated the modifiable walking patterns for the following planned footsteps. The effectiveness of the proposed method was demonstrated through simulation and experiment using the small-sized humanoid robot platform DARwIn-OP. As a result, the proposed method enables a humanoid robot to arrive at the goal without obstacle collisions following the planned footsteps in real-time.

Acknowledgements

This work was supported by the new faculty research fund of Ajou University.

References

- [1] M. Yagi and V. Lumelsky, "Biped robot locomotion in scenes with unknown obstacles," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1999, pp. 375-380.
- [2] E. T. Wong and R. Jarvis, "Real time obstacle detection and navigation planning for a humanoid robot in an indoor environment," in *Proc. IEEE Conf. Robot. Autom. Mechatron.*, 2004, pp. 693-698.
- [3] J. Chestnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins, and T. Kanade, "Footstep planning for the honda asimo humanoid," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 629-634.
- [4] J. Chestnutt, P. Michel, K. Nishiwaki, J. Kuffner, and S. Kagami, "An intelligent joystick for biped control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 860-865.
- [5] Z. Xia, J. Xiong, and K. Chen, "Parameter self-adaptation in biped navigation employing nonuniform randomized footstep planner," *Robotica*, vol. 28, no. 6, pp. 929-936, Oct. 2010.
- [6] Z. Xia, J. Xiong, and K. Chen, "Global navigation for humanoid robots using sampling-based footstep planners," *IEEE/ASME Trans. Mechatron.*, vol. 16, no. 4, pp. 716-723, Aug. 2011.
- [7] N. Perrin, O. Stasse, L. Baudouin, F. Lamiriaux, and E. Yoshida, "Fast humanoid robot collision-free footstep planning using swept volume approximations," *IEEE Trans. Robot.*, vol. 28, no. 2, pp. 427-439, Apr. 2012.
- [8] Y.-D. Hong, Y.-H. Kim, J.-H. Han, J.-K. Yoo, and J.-H. Kim, "Evolutionary multiobjective footstep planning for humanoid robots," *IEEE Trans. Syst. Man. Cybern. C, Appl. Rev.*, vol. 41, no. 4, pp. 520-532, Jul. 2011.
- [9] Y. -D. Hong and J. -H. Kim, "An evolutionary optimized footstep planner for the navigation of humanoid robots," *Int. J. Humanoid Robot.*, vol. 9, no. 1, Mar. 2012.
- [10] Y.-J. Kim, J.-H. Kim, and D.-S. Kwon, "Evolutionary programming based univector field navigation method for fast mobile robots," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 31, no. 3, pp. 450-458, Jun. 2001.
- [11] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Yokoi, and H. Hirukawa, "A realtime pattern generator for biped walking," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, vol. 1, pp. 31-37.
- [12] B.-J. Lee, D. Stonier, Y.-D. Kim, J.-K. Yoo, and J.-H.

- Kim, "Modifiable walking pattern of a humanoid robot by using allowable ZMP variation," *IEEE Trans. Robot.*, vol. 24, no. 4, pp. 917-925, Apr. 2008.
- [13] Y.-D. Hong, B.-J. Lee, and J.-H. Kim, "Command state-based modifiable walking pattern generation on an inclined plane in pitch and roll directions for humanoid robots," *IEEE/ASME Trans. Mechatron.*, vol. 16, no. 4, pp. 783-789, Aug. 2011.
- [14] Y.-D. Hong and J.-H. Kim, "3-D command state-based modifiable bipedal walking on uneven terrain," *IEEE/ASME Trans. Mechatron.*, vol. 18, no. 2, pp. 657-663, Apr. 2013.
- [15] O. Michel, "Cyberbotics Ltd.WebotsTM: Professional mobile robot simulation," *Int. J. Adv. Robot. Syst.*, vol. 1, no. 1, pp. 39-42, 2004.

Young-Dae Hong received B.S., M.S., and Ph.D. degrees in electrical engineering from KAIST, Daejeon, Korea, in 2007, 2009, and 2013, respectively. Since 2014, he has been with the Department of Electrical and Computer Engineering at Ajou University, Suwon, Korea, where he is currently an

Assistant Professor. His current research interests include humanoid robotics, especially bipedal walking pattern generation, footstep planning, multi-/single-objective evolutionary optimization, and biologically inspired control.