2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Macau, China, November 4-8, 2019

Autonomous Safe Locomotion System for Bipedal Robot Applying Vision and Sole Reaction Force to Footstep Planning

Yuki Omori, Yuta Kojio, Tatsuya Ishikawa, Kunio Kojima,
Fumihito Sugai, Yohei Kakiuchi, Kei Okada and Masayuki Inaba

***Abstract*— Humanoid robots are expected to conduct tasks on behalf of humans in places such as a disaster scattered environment. Although humanoid robots have potentials to walk on uneven ground unlike wheeled robots, it is difficult to reach a given destination without falling down based on only visual information. In this paper, to reach the destination safely, we propose the autonomous safe locomotion system applying vision and sole reaction force to the footstep planning. Considering force information in addition to visual information, the robot can plan a path avoiding unstable footholds. The planned path is safer than a path which is planned based on only visual information. In our system, the robot checks if the foothold is safe or not by the foothold ascertainment motion. In addition to that, the robot saves the results of the motion to the database with the foothold label given by the visual classifier. To judge foothold safety, stiffness of the foothold is estimated from the reaction force and stepping amount. We propose the system considering these requirements for safe locomotion for bipedal robots and show experimental results using a real bipedal robot CHIDORI.**

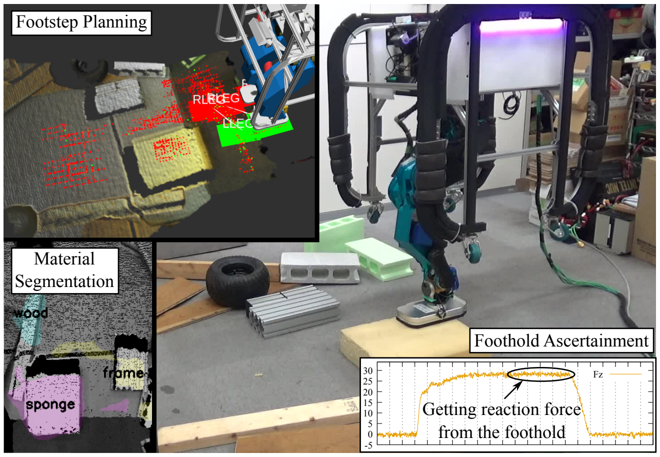


Fig. 1: Life-sized bipedal robot CHIDORI estimating robustness of a soft foothold by the foothold ascertainment motion while reaching a goal.

I. INTRODUCTION

Humanoid robots are expected to conduct tasks on behalf of humans in places such as a disaster environment. These kind of environments tend to include uneven ground and robots have to step over there to reach the goal where they conduct tasks. As studies of path planning for bipedal robots, Kuffner *et al.* [3] proposed a footstep planning method with A* search algorithm [4] and this method has been widely implemented to footstep planning. Path planning for bipedal robots commonly bases on only visual information [5], [6]. To make an accurate environmental map for footstep planning, LiDAR is widely used as a vision sensor [7]. RGBD sensors are also used to make an environmental map fast for footstep planning and navigation to overcome that LiDAR takes much time to make a map [8], [9].

As footstep planning methods on uneven ground, height information is used not only for placing footsteps but for being implemented to cost functions in a planning solver. Moreover, Kanoulas *et al.* [10] proposed a footstep planning method considering paraboloid in addition to plane as a foothold based on visual information. Although these studies have considered only environmental shape information to plan footsteps, robots have to obtain physical properties of footholds to reach a goal without falling down on the way to the goal. Especially in a disaster environment, bipedal robots have to step over footholds which are not fixed to the environment and deform when they step on. To obtain another foothold information in addition to the shape from a vision sensor, Brandao *et al.* [1] proposed a method for friction estimation of the ground. However, obtaining physical properties from vision is still limited.

TABLE I: Classification of previous studies of bipedal walking with environment information.

Approach	Vision		Force Sense	
	Shape	Material	Shape	Material
Common	✓			
Brandao's [1]	✓	△		
Wiedebach's [2]	✓		✓	
Ours	✓			✓

✓acheived △ partially acheved

Humans use both visual and force information when they step over uneven and unstable footholds. They visually explore footholds to step on and ascertain them when the footholds seem unstable. Bipedal robots can also use these same strategies as humans. Therefore, obtaining material properties of footholds not only from vision sensors but also from force sensors is a potential solution of safe footstep planning for bipedal robots.

In the fields of bipedal robots, balance control with ZMP (Zero Moment Point) has been widely researched to avoid falling down [11]–[13]. These balance control methods use wrench information from force sensors on the sole. The information can be also used for obtaining material properties of footholds. As studies of contact behavior with an environment, Nozawa *et al.* [14] proposed a balance control method with updating reference force in the case such as opening

978-1-7281-4003-2/19/$31.00 ©2019 IEEE

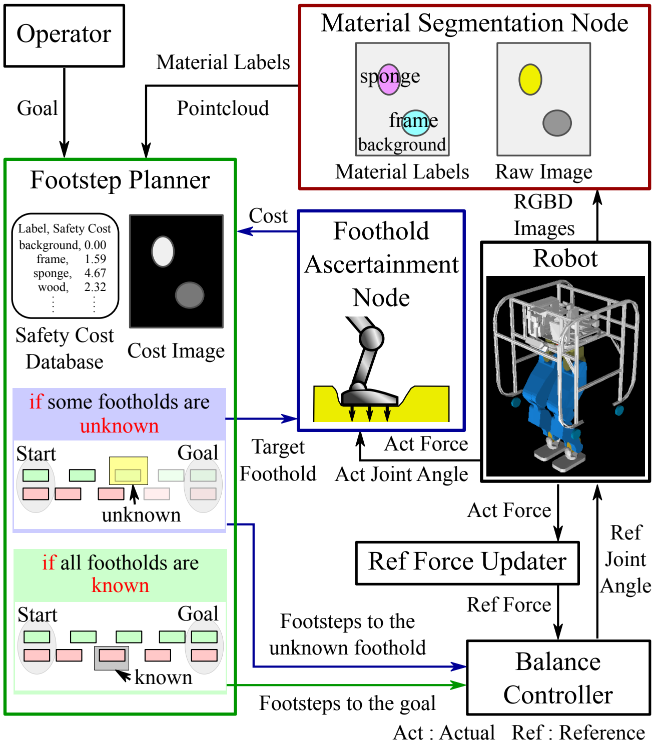


Fig. 2: Autonomous safe locomotion system overview. The system consists of four components: Foothold Ascertainment Node (Blue), Footstep Planner (Green), Material Segmentation Node (Red), and Controller.

a door by hand. This method is also applied to calculate buoyancy in water [15].

As studies of obtaining information from force sensors on the sole for walking balance control, Widebach *et al.* [2] proposed a method obtaining local shape information such as edge of a foothold. This method enables bipedal robots to walk on line and point. Moreover, Hoepflinger *et al.* [16] proposed a method of ascertaining foothold robustness for legged robot. They integrated vision and force information to estimate foothold robustness. As another related work, Guadarrama *et al.* [17] used original tactile sensors on the soles of their humanoid robot. The tactile sensors have both force sensors and proximity sensors for the robot to detect small obstacles before stepping-on and avoid them.

In this paper, we propose a method to obtain stiffness as material information of a foothold from force sensors on the sole by the foothold ascertainment motion. We also propose the system using the information of stiffness in the footstep planner. Therefore, both shape and material information are taken into account in the footstep planning. Table I shows classification of previous studies and our method. From next section, the proposed system and each component of the system are described. Lastly, experimental results are shown as shown in Fig.1 which our bipedal robot CHIDORI ascertained footholds to step on while reaching the goal.

II. Autonomous Safe Locomotion System

In this section, we describe the autonomous safe locomotion system applying visual and sole reaction force information to the footstep planning for bipedal robots (Fig.2). Implementing not only shape information from vision sensors, but also material information from force sensors, a robot can reach a goal safely without falling down. The proposed system consists of broadly four components: *Foothold Ascertainment Node*, *Footstep Planner*, *Material Segmentation Node*, and *Controller*. The brand new component is foothold ascertainment node for obtaining stiffness of a foothold by force sensors on the swing leg. The acquired stiffness is used to calculate foothold's safety cost which is implemented to the footstep planning.

In this system, an operator gives goal position to the footstep planner via GUI. The material segmentation node also gives material labels generated by the material classifier. Raw images are given by RGBD sensor mounted on the robot. After receiving the goal, pointcloud of the environment and the material labels, the footstep planner plans a path to the goal considering shape and material of the footholds. While planning footsteps, safety cost which is related to robustness of the foothold is considered. The safety cost of the foothold is registered in the safety cost database with its material label given by the material classifier. If all footholds planned to step-on have material labels which are registered in the database, the robot starts to go to the goal. On the other hand, if the planned path includes an unknown foothold which is not registered in the database, the robot goes to right before the unknown foothold and ascertain its robustness by the foothold ascertainment motion. After calculating the safety cost by stiffness of the foothold, the footstep planner plans to the goal again. This loop continues until the robot reaches the goal. This is the whole flow of the autonomous safe locomotion system.

Each component is described in following sections. Advantages of this proposed system are

- Safe Footstep Planning:
  A robot can choose safer footholds by using material information of the foothold from force sensors in addition to its shape information from vision sensors.
- Efficient System:
  Saving stiffness of the foothold with its label in the database, the planner can reuse stiffness information and omit the motion for registered materials.

III. Foothold Robustness Estimation

In this section, we describe motion generation for foothold ascertainment to estimate foothold robustness and to use force information while planning footsteps to the goal. Next, to keep balance during the motion, balance control methods are described. Then, reaction force gained from a target foothold is given, stiffness of the foothold is calculated from the reaction force and the stepping amount. Lastly, safety cost and its calculation method from given reaction force are introduced. This section corresponds to the blue frame node (*Foothold Ascertainment Node*) in the Fig.2.

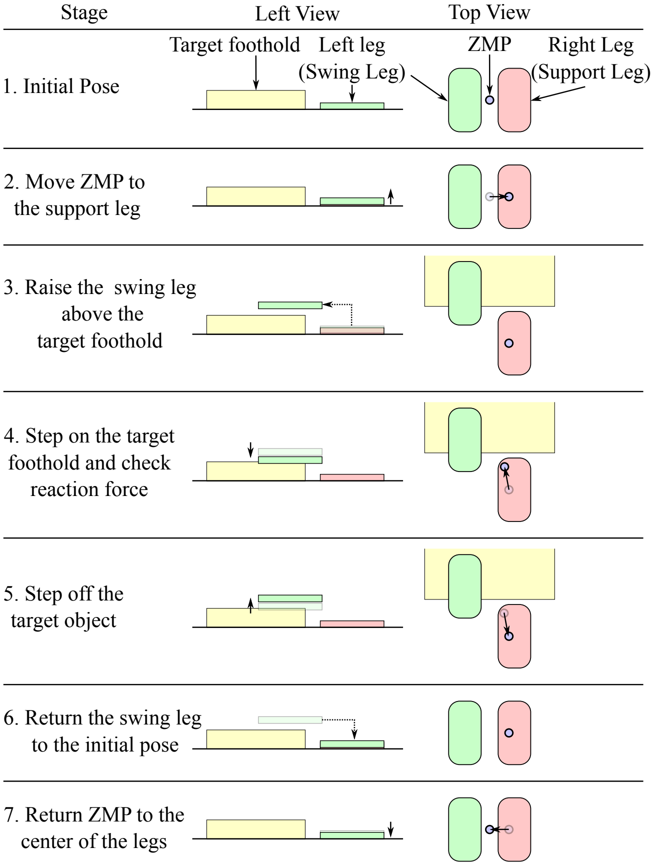

Stage
Left View
Top View
Target foothold
Left leg (Swing Leg)
ZMP
Right Leg (Support Leg)
1. Initial Pose
2. Move ZMP to the support leg
3. Raise the swing leg above the target foothold
4. Step on the target foothold and check reaction force
5. Step off the target object
6. Return the swing leg to the initial pose
7. Return ZMP to the center of the legs

Fig. 3: Steps of foothold ascertainment motion and ZMP position of each stage. This is the case when a robot ascertains a forward foothold by the left leg.

A. Motion Generation for Foothold Ascertainment

To ascertain stiffness of a foothold, the foothold ascertainment motion different from the walking motion is generated. While ascertaining a foothold, a robot has to avoid falling down. The motion is composed of 7 stages. Motion and ZMP of each stage is shown in Fig.3. In Fig.3, both legs are aligned at initial pose, however in real case, they are not always aligned because the motion is taken on the way to the goal. Target coordinate of the swing leg to the foothold is set on a part of original planned path to the goal given by the footstep planner.

After deciding the target coordinate to step on the target foothold, the ZMP is moved into the support polygon of the legs at stage 2. Then, the swing leg is moved just above the target coordinate (x and y axis are the same as the target coordinate) at stage 3. From stage 4 to 5, the robot steps on and off the target foothold. Here the robot gets reaction force from the target foothold and estimates how much stable it is. Concretely, stiffness of the target foothold is calculated from the reaction force and the stepping amount which is calculated as the relative coordinates between the swing leg and the support leg. The remaining stages are reverse motions of stage 2 and 3.

B. Balance Control for Foothold Ascertainment Motion

While sensing reaction force from the target foothold, the robot must keep its balance. We describe three methods to keep the balance during the motion, pros and cons.

1) Reference Position Command:
Setting reference position and sensing reaction force from a target foothold, stiffness of the foothold can be estimated. In this method, ZMP is in the support polygon of the support leg while sensing reaction force. That is safe when the robot steps on a soft foothold, because the ZMP keeps in the support polygon. However, when the robot steps on a hard one, the robot falls down, because the reaction force is too large that ZMP goes out of the support polygon.

2) Reference Force Command:
Setting reference force and calculating stepping amount is another method to estimate stiffness of the foothold. In this method, the robot keeps its balance considering the reference force by Eq.1, where x_G is position of the center of gravity, Δx_G is update amount of x_G, x_{swg} is the end coordinate of the swing leg, f_{ref} is reference force of the reaction force, M is mass of the robot, and g is the gravitational acceleration [18]. This foothold ascertainment motion is quasi-static, so the center of gravity accords the ZMP. Applying this method, that is safe when the robot steps on a hard foothold, because the robot can gain the reaction force immediately. On the other hand, when the robot steps on a soft one, the robot falls down, because it takes much time to gain the reference force from the foothold or the robot cannot gain the reference force from it. When the robot cannot gain the reference force from the foothold, it keeps pointing the swing leg down and loses the balance.

$$x_G + \Delta x_G = \frac{x_G(Mg - f_{ref}) + x_{swg} f_{ref}}{Mg} \tag{1}$$

3) Reference Position Command with Reference Force Update:
Based on the above, to deal with both hard and soft footholds, setting reference position and updating reference force is assumed better. In this method, if large reaction force is gained, the reference force increases rapidlly and the robot puts its weight on the foothold a lot following Eq.1. On the other hand, if small reaction force is gained, the referecne force does not become so large that the robot puts its weight on the foothold a little. Thus, this method can deal both hard and soft footholds, so this balance control method is applied to the foothold ascertainment motion.

To apply the third method, some control parameters are adjusted. Eq.2 is reference force update law, where f_{ref} is reference force, f_{act} is actual force, and K is constant positive gain [14].

$$f_{ref}^{k+1} = f_{ref}^{k} + K(f_{act}^{k} - f_{ref}^{k}) \tag{2}$$

Gain K in Eq.2 and stepping time should be adjusted to avoid falling down during the motion. When K is small, it takes much time to converge to reference force. On the other hand, when it is too large, that causes vibration when the leg contacts the foothold. Stepping-on time should be adjusted with K. If stepping-on motion is fast with small K, that makes a robot fall down because the reference force cannot follow as fast as the motion.

On the other hand, the gain K is not used for stepping-off motion. The robot steps off and returns ZMP to the support leg's ZMP origin at the same time, which prevents failure cases that the robot keeps putting its weight on the foothold and the reference ZMP does not return to the support leg's ZMP origin.

Moreover, for sensing the reaction force, waiting time and checking time between stepping-on and stepping-off the foothold are set. Each time of these four sequences is set to 5 seconds as shown in Fig.4. Waiting time is for covergence of the reaction force F_z and the end coordinates P_z. Checking time is for mesuring converged F_z and P_z.

While balancing by the reference force control, reference ZMP limit is set to avoid putting much weight on the foothold. This limit works when the robot gains much reaction force form the foothold. If the maximum reference force f_{ref}^{max} when the robot steps on the hard foothold and maximum stride x_s^{max} are given, maximum distance of ZMP $\Delta\hat{x}_Z^{max}$ can be calculated as Eq.3.

$$\Delta\hat{x}_Z^{max} = \left| \frac{f_{ref}^{max} x_s^{max}}{Mg} \right| \tag{3}$$

The adequate ZMP limit x_Z^{lim} must be set satisfing Eq.4, where x_r^{mgn} is distance between the original reference ZMP and rear edge of the support polygon, and x_f^{mgn} is distance between the original reference ZMP and front edge of the support polygon. Both x_r^{mgn} and x_f^{mgn} are constant positive. Left inequality guarantees that the robot can keep balance when $\Delta\hat{x}_Z$ exceeds the ZMP limit x_G^{lim}. Right inequality guarantees that the robot does not fall down due to collapse of the foothold, which means the ZMP is inside of the support leg durint the foothold ascertainment motion. Eq.3 and Eq.4 can be applied when the robot steps on backward or sideways if the margins are replaced.

$$\Delta\hat{x}_Z^{max} - x_r^{mgn} < x_G^{lim} < x_f^{mgn} \tag{4}$$

Regarding the stepping amount, if the amount is too large, the robot cannot keep balance even controlled with the reference force updater. However, the amount is too small, the robot cannot distinguish stiffness between a soft unsafe foothold and a hard safe foothold. Therefore, original reference stepping amount is set and the robot stops stepping-on motion if the difference between the actual stepping amount and the reference stepping amount becomes large.

C. Foothold Robustness Estimation by Stiffness

Stiffness of a foothold is closely related to its robustness when a robot steps on the foothold. Of course there are a lot of physical properties to consider foothold robustness, in this paper stiffness is chosen as the most important parameter and used it to estimate foothold robustness. Fig.4 shows the relative end coordinates between the swing leg and the support leg, and it also shows reaction force from a target foothold. This plot consists of four parts as mentioned above.

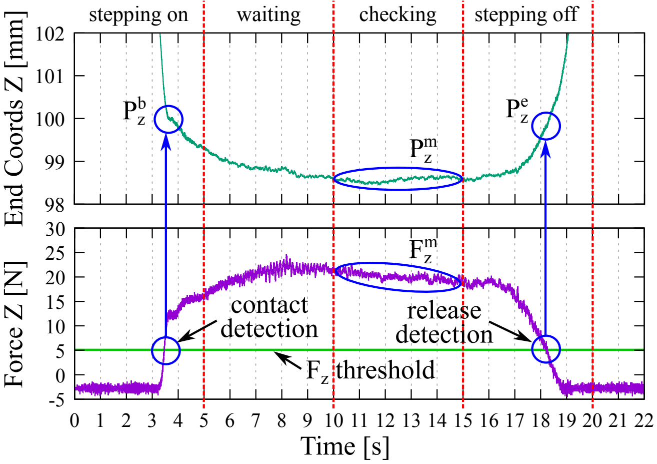


Fig. 4: Reaction force of a target foothold and the relative end coordinates between the swing leg and the support leg. These values are used to calculate stiffness of the foothold. Waiting time is for convergence of the reaction force and the end coordinates.

Stiffness k is calculated as Eq.5, where P_z^b, P_z^e, P_z^m, and F_z^m are shown in Fig.4. F_z^{th} is threshold to detect contact to the target foothold. P_z^b and P_z^e are P_z when the reaction force F_z become larger and lower than F_z^{th}.

$$k = \frac{\Delta F_z}{\Delta P_z} = \frac{F_z^m - F_z^{th}}{\frac{1}{2}(P_z^b + P_z^e) - P_z^m} \tag{5}$$

Stiffness of the target foothold calculated by the above method is not accurate stiffness of the foothold, because it includes rubber stiffness on the sole and stiffness of the leg joints. In other words, stiffness is subjective value for the robot. However, obtaining accurate stiffness by the leg of a bipedal robot is difficult and subjective stiffness is sufficient information for a robot. This is because if accurate stiffness of the foothold is obtained, the robot has to consider both stiffness of the foothold and its own stiffness when the robot steps on the foothold.

After calculating stiffness of the target foothold, safety cost of the foothold is estimated to be used in the footstep planner. Safecy cost c_s is calculated as Eq.6, where w and k_{off} are parameters for adjusting range. Example parameters are shown in Table II. Using logarithm, large stiffness can be dealt. Safety cost of a foothold with large stiffness is calculated as small value and safety cost of a foothold with small stiffness is calculated as large value.

$$c_s = \max\{0, -w \ln \frac{k}{k_{off}}\} \tag{6}$$

TABLE II: Example of safety cost parameters

w	5.0
k_{off}	4.5e+4

IV. Safe Footstep Planning

In this section, the method of planning footsteps with stiffness information in addition to visual information is described. Firstly, *safety cost* and *safety cost database* are introduced as follows.

- Safety Cost:
  Safety cost is value to be implemented to the footstep planning, and it is calculated from foothold stiffness. Default value of the *safety cost* is 0 and it is updated when a robot practise the foothold ascertainment motion. *Safety cost* refers how dangerous when the robot steps on the foothold.
- Safety Cost Database:
  Safety cost database is database to save material labels of footholds and *safety cost* of them. The database is referenced while generating a cost image form material labels. The cost image is referenced while planning for nodes to obtain safety costs of corresponding step coordinates.

These two are important concepts in the system, because adding safety cost, the planned path becomes safer. Moreover, saving results of the foothold ascertainment motion with its material label, efficient footstep planning comes true. Here a method implementing safety cost to the footstep planning is described and its result are compared with the result of existing method. Finally, the system correspondence is described in the case where the planned path includes unknown footholds; not registered its safety cost in the safety cost database. This section corresponds to the red frame node (*Material Segmentation Node*) and the green frame node (*Footstep Planner*) in the Fig.2.

A. Foothold Recognition Based on Material Segmentation

Humans recognize foothold material and estimate its robustness based on their experiences. To memorize safety costs with material labels to reuse them, a material classifier for labeling material is required. The desirable classifier is made by unsupervised learning, because the classifier has scalability to deal unknown materials in future. However, meaningful segmentation for applying foothold robustness by unsupervised learning has not been proposed. In this system, the classifier is prepared by supervised learning in advance and the classifier labels limited objects. The flow of making a cost image from a raw image is shown in Fig.5. The raw image is obtained by RGBD cameras on the robot and converted to material labels by the material classifier. The classifier is made by FCN-8s (Fully Convolutional Networks) [19]. Then the material labels are converted to the cost image referencing the safety cost database. The cost image has safety cost information of each pixel and it is used in the footstep planner.

B. Safety Cost Implementation to Footstep Planning

A* search algorithm is applied to the footstep planning. To apply the algorithm, area which the swing leg can step are discretized [3]. Three dimensional footstep planning is realized by projecting two dimensional footsteps to point-cloud obtained by vision sensors and then judging if the projected footsteps can be supported by the environment. In the algorithm, which node should be opened next is decided by its cost. A node with the least cost of all is opened first. Each node has information as Eq.7, where p is coordinate, q is posture, l is leg, and s is safety cost. The cost is the sum of heuristic function and cost function as Eq.8, where $g(N)$ is cost function, and $h(N)$ is heuristic function. Safety cost is implemented to calculate both heuristic function and cost function to consider stiffness of foothold.

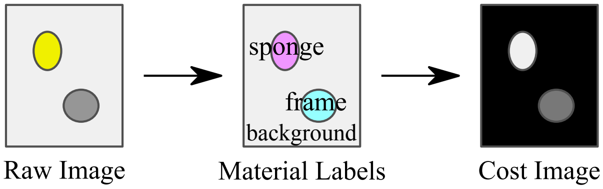


Fig. 5: Foothold recognition flow. Raw image from a RGBD camera is converted to material labels by a classifier. Then the material labels are converted to a cost image referring to the safety cost database.

$$N = (p, q, l, s) \tag{7}$$

$$f(N) = g(N) + h(N) \tag{8}$$

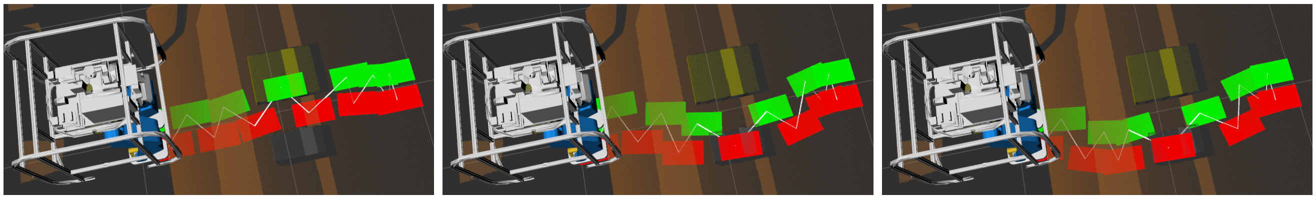
Generally, only step cost is used in cost function. Step cost means number of steps and corresponds to depth of a target node. In this study, the cost function is extended as Eq.9, where w_s is weight for safety cost, and s is safety cost of the target node N. N_p is a parent node of the target node. The second term is step cost and the third term is weighted safety cost. Implementing safety cost to the cost function, low safety cost nodes take priority and the planned path avoids nodes with high safety cost. Namely, the planned path excepts dangerous footholds such as soft objects.

$$g(N) = g(N_p) + 1 + w_s s \tag{9}$$

Heuristic function is also extended to make calculation time shorter as Eq.10, where $l_{\max}$ is maximum length of one step. The first term is step cost and the second term is weighted safety cost. At the first term, number of remaining steps is calculated when a robot straightly goes to the goal from the target node. At the second term, the sum of safety costs of remaining steps is calculated. Heuristic function is for making exploring time shorter. Therefore, the function does not change the path dramatically.

$$h(N, N_g) = \frac{|p - p_g|}{l_{\max}} + w_s \sum_{k}^{steps} s_k \tag{10}$$

Although these functions prior to plan a safer path, they sometimes a plan path which includes a dangerous foothold when there are only dangerous footholds in the environment. Safety cost is continuous value, however we threshold should be set to exclude footholds where a robot cannot step on.



(a) $g(n)$: Step Cost
$h(n)$: Step Cost
Number of Steps : 12
Calculation Time : 0.0983[s]
Conventional planning result of footsteps to the goal. The planned path includes a foothold with high safety cost.

(b) $g(n)$: Step Cost + Safety Cost
$h(n)$: Step Cost
Number of Steps : 12
Calculation Time : 20.1[s]
Planning results of footsteps to goal considering safety cost. The planned path is changed to avoid a yellow foothold with high safety cost.

(c) $g(n)$: Step Cost + Safety Cost
$h(n)$: Step Cost + Safety Cost
Number of Steps : 12
Calculation Time : 18.2[s]
The planned path is almost same as before. The advantage implementing safety cost to heuristic function is making calculation time shorter.

Fig. 6: Comparison of paths planned by existing method and proposing method in a simulation environment. Implementing safety cost to cost function, the planner avoids unsafe yellow foothold.

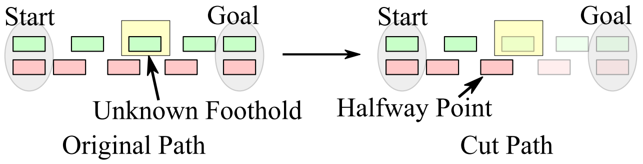


Fig. 7: The original path cut right before the unknown yellow foothold.

Planned path of existing method and proposing method are shown in Fig.6. The differences of them are cost function and heuristic function. We prepared simulation environment where a gray object with low safety cost and a yellow object with high safety cost are placed on the floor. A robot should choose one of them as foothold to reach the goal. While considering only step cost, the path includes yellow foothold with large area. However, considering both step cost and safety cost, the path is changed to include the gray foothold which is safer than the yellow foothold.

C. Correspondence to Unknown Foothold

As safety cost is registered to the safety cost database when a robot takes foothold checking motion and senses reaction force from the foothold, safety costs of some material labels have not registered in the database. These footholds which have not registered with safety cost in the database are called unknown footholds. In the situation where there are some unknown footholds in the environment, the planner plans footsteps considering unknown footholds have certain safety cost. Then the planner judges if the planned path includes unknown footholds or not. If the planned path includes an unknown foothold, it is cut right before the unknown foothold and set there as a halfway point. The robot will check the unknown foothold after it arrives the halfway point. Fig.7 shows the cut path right before an unknown yellow foothold.

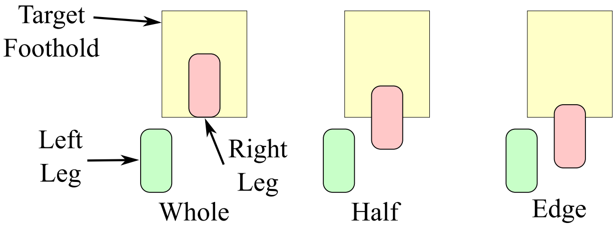


Fig. 8: 3 types of contact area in the foothold ascertainment experiment

V. Experiments

A. Life-Sized Bipedal Robot CHIDORI

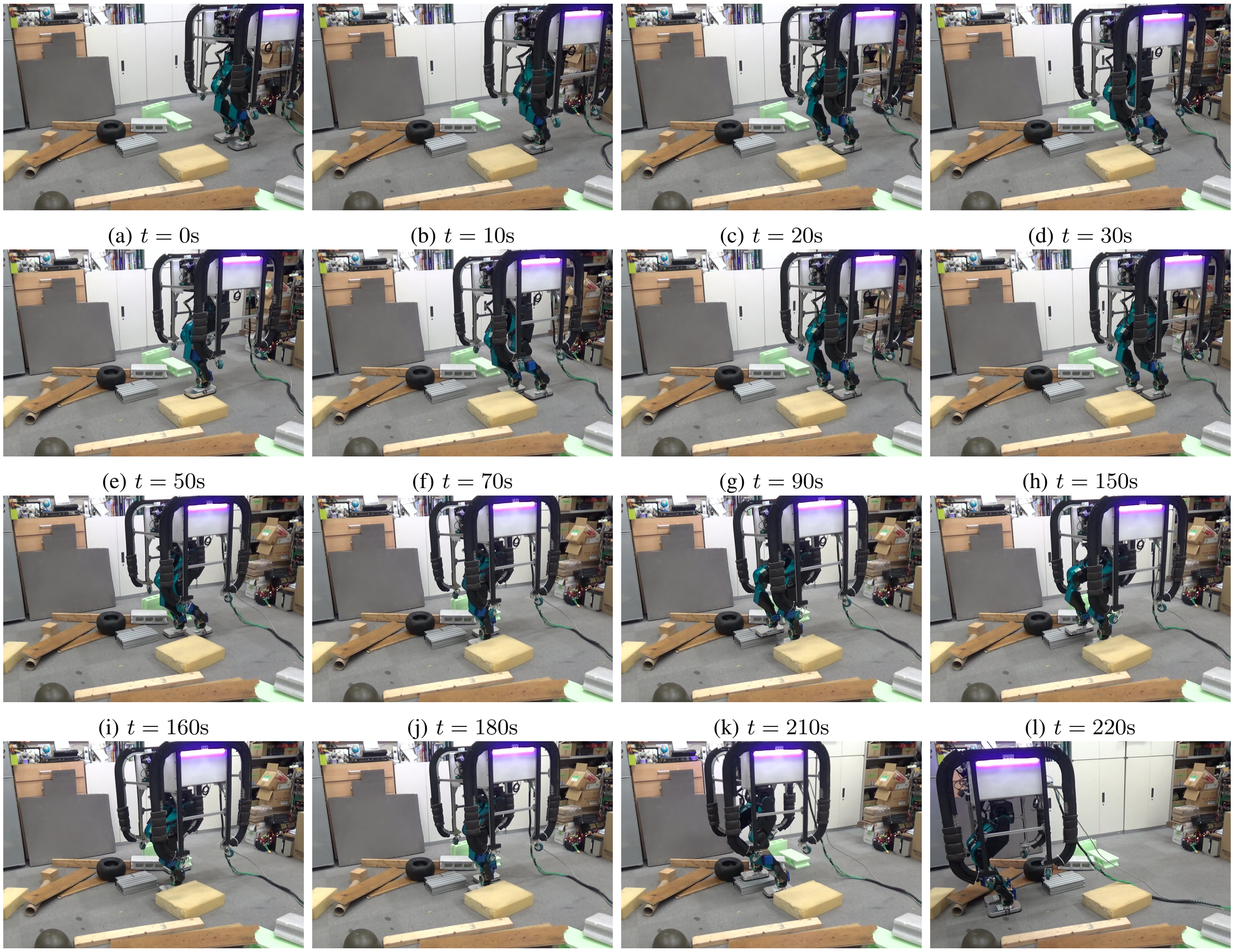
We validated the proposed system by implementing it to the life-sized bipedal robot CHIDORI [20] shown in Fig.1. CHIDORI is 1.3[m] tall and the weight is 67 [kg] including protection frames. CHIDORI has 6 axis force/torque sensors on each foot, which is used to obtain reaction force from a foothold. On the sole, rubber is attached to adapt floor and footholds. CHIDORI has two RGBD cameras (ASTRA [21]) on the waist. One is mounted downwards and the other one is mounted forward diagonally.

B. Foothold Ascertainment Experiment

Before conducting integration experiment, stiffness is calculated by the foothold ascertainment motion for 3 materials (metal frame, board and sponge) at 3 different area shown in Fig.8. The results shown in Table III means that naturally stiffness differs among materials, it also differs in even a same material at different area. Although stiffness should be modified according to where the robot steps on in the foothold ascertainment motion, raw values are used in the integration experiment. This is because the robot compares only metal frame and sponge on the integration experiment and the difference of the stiffness between them is large.

C. Integration Experiment

The proposed system was implemented to CHIDORI and integration experiment was conducted. Metal frame as a safe foothold, sponge as a dangerous foothold, and other materials



(a) $t = 0$s (b) $t = 10$s (c) $t = 20$s (d) $t = 30$s

(e) $t = 50$s (f) $t = 70$s (g) $t = 90$s (h) $t = 150$s

(i) $t = 160$s (j) $t = 180$s (k) $t = 210$s (l) $t = 220$s

(m) $t = 230$s (n) $t = 260$s (o) $t = 300$s (p) $t = 310$s

Fig. 9: The robot went to the soft yellow foothold (a-c), checked it and gained reaction force (d-f). Then replanned footsteps (g) and went to the metal foothold (h-j). The metal foothold was also unknown, so the robot ascertained its reaction force (k-m). Finally, the robot replanned footsteps (n) and reached the goal (o-p).

TABLE III

Foothold	Contact Area	ΔF_z[N]	ΔP_z[mm]	k[N/m]
Metal Frame	Whole	97.5	3.58	2.72e+4
	Half	94.2	4.22	2.00e+4
	Edge	75.8	4.54	1.67e+4
Board	Whole	87.7	6.21	1.41e+4
	Half	88.0	7.67	1.02e+4
	Edge	77.6	7.81	9.95e+3
Sponge	Whole	87.8	8.97	9.79e+3
	Half	65.9	10.2	6.49e+3
	Edge	37.8	11.9	3.18e+3

were prepared to make a scattered environment. All footholds information was not registered in the safety cost database in advance. The classifier can add labels to footholds.

Firstly, the robot planned path to the goal and the path includes the sponge as a foothold because the shape is adequate to support a leg and the area of the sponge is the largest in the environment. The sponge was not registered in the safety cost database, and therefore, the robot stopped walking right before the sponge and obtained its stiffness. Safety cost of the sponge was calculated from obtained stiffness and it was registered to the database. Then the robot replanned a path to the goal considering safety cost of the sponge (safety cost of the metal frame was still unknown at this time). The planned path includes the metal frame because the safety cost of the sponge was large it was on the way to the goal. However, the safety cost of the metal frame was still unknown, the robot went to right before the metal frame and checked it. After calculating safety cost of the metal frame, the robot replanned a path to the goal considering safety costs of the sponge and the metal frame. The planned path included only the frame as a foothold and the frame was known, the robot stepped on it and finally went to the goal. Captures of this integration experiment are shown in Fig.9.

The results of the foothold ascertainment motion on the way to the goal are shown in Table IV. ΔF_z, ΔP_z and k

TABLE IV: Results of the foothold ascertainment motion on the way to the goal. Safety cost c_s is calculated using parameters of Table II.

Foothold	ΔF_z[N]	ΔP_z[mm]	k[N/m]	c_s
Sponge (Soft)	18.1	2.24	8.08e+3	8.59
Metal Frame (hard)	107	3.82	2.81e+4	2.35

are obtained by Eq.5. Safety cost c_s is calculated Eq.6 using parameters of Table II.

VI. Conclusion

In this paper, we proposed the autonomous safe locomotion system and described each components of the system. By the foothold ascertainment motion, a bipedal robot can avoid a dangerous foothold which looks adequate to step on because of its shape, though the foothold ascertainment motion currently takes much time. The foothold ascertainment motion can cope with foothold collapse during the motion. However, the motion cannot judge if the foothold is safe when whole weight puts on it. The current motion is quasi-static, but dynamic foothold ascertainment motion is required to judge that for the safer system. The footstep planner avoids unsafe footholds considering both step cost and safety cost. It also currently takes much time. The material classifier can match materials in the planned path and the safety cost database, and it gives stiffness information to the footstep planner. That can omit the time consuming foothold ascertainment motion it the target foothold has already been ascertained.

References

[1] Martim Brandao, Kenji Hashimoto, and Atsuo Takanishi. Friction from vision: A study of algorithmic and human performance with consequences for robot perception and teleoperation. In *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, pp. 428–435. IEEE, 2016.

[2] Georg Wiedebach, Sylvain Bertrand, Tingfan Wu, Luca Fiorio, Stephen McCrory, Robert Griffin, Francesco Nori, and Jerry Pratt. Walking on partial footholds including line contacts with the humanoid robot atlas. In *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, pp. 1312–1319. IEEE, 2016.

[3] James J Kuffner Jr, Koichi Nishiwaki, Satoshi Kagami, Masayuki Inaba, and Hirochika Inoue. Footstep planning among obstacles for biped robots. In *IROS*, pp. 500–505, 2001.

[4] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, Vol. 4, No. 2, pp. 100–107, 1968.

[5] Philipp Michel, Joel Chestnutt, Satoshi Kagami, Koichi Nishiwaki, James Kuffner, and Takeo Kanade. Gpu-accelerated real-time 3d tracking for humanoid locomotion and stair climbing. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pp. 463–469. IEEE, 2007.

[6] Joel Chestnutt, Koichi Nishiwaki, James Kuffner, and Satoshi Kagami. An adaptive action model for legged navigation planning. In *Humanoid Robots, 2007 7th IEEE-RAS International Conference on*, pp. 196–202. IEEE, 2007.

[7] Ryohei Ueda, Shunichi Nozawa, Kei Okada, and Masayuki Inaba. Biped humanoid navigation system supervised through interruptible user-interface with asynchronous vision and foot sensor monitoring. In *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*, pp. 273–278. IEEE, 2014.

[8] Daniel Maier, Armin Hornung, and Maren Bennewitz. Real-time navigation in 3d environments based on depth camera data. In *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on*, pp. 692–697. IEEE, 2012.

[9] Péter Fankhauser, Michael Bloesch, Diego Rodriguez, Ralf Kaestner, Marco Hutter, and Roland Y Siegwart. Kinect v2 for mobile robot navigation: Evaluation and modeling. In *2015 International Conference on Advanced Robotics (ICAR)*, pp. 388–394. IEEE, 2015.

[10] Dimitrios Kanoulas, Chengxu Zhou, Anh Nguyen, Georgios Kanoulas, Darwin G Caldwell, and Nikos G Tsagarakis. Vision-based foothold contact reasoning using curved surface patches. In *Humanoid Robotics (Humanoids), 2017 IEEE-RAS 17th International Conference on*, pp. 121–128. IEEE, 2017.

[11] Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kazuhito Yokoi, and Hirohisa Hirukawa. A realtime pattern generator for biped walking. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, Vol. 1, pp. 31–37. IEEE, 2002.

[12] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Resolved momentum control: humanoid motion planning based on the linear and angular momentum. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 2, pp. 1644–1650, Oct. 2003.

[13] Kazuhito Yokoi, Fumio Kanehiro, Kenji Kaneko, Shuuji Kajita, Kiyoshi Fujiwara, and Hirohisa Hirukawa. Experimental study of humanoid robot hrp-1s. *The International Journal of Robotics Research*, Vol. 23, No. 4-5, pp. 351–362, 2004.

[14] Shunichi Nozawa, Iori Kumagai, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Humanoid full-body controller adapting constraints in structured objects through updating task-level reference force. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 3417–3424. IEEE, 2012.

[15] Yuta Kojio, Tatsushi Karasawa, Kunio Kojima, Ryo Koyama, Fumihito Sugai, Shunichi Nozawa, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Walking control in water considering reaction forces from water for humanoid robots with a waterproof suit. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 658–665, 2016.

[16] Mark A Hoepflinger, Marco Hutter, Christian Gehring, Michael Blösch, and Roland Siegwart. Unsupervised identification and prediction of foothold robustness. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 3293–3298. IEEE, 2013.

[17] J Rogelio Guadarrama-Olvera, Florian Bergner, Emmanuel Dean, and Gordon Cheng. Enhancing biped locomotion on unknown terrain using tactile feedback. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, pp. 1–9. IEEE, 2018.

[18] Kensuke Harada, Shuuji Kajita, Kenji Kaneko, and Hirohisa Hirukawa. Zmp analysis for arm/leg coordination. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, Vol. 1, pp. 75–81. IEEE, 2003.

[19] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[20] Tatsuya Ishikawa, Yuta Kojio, Kunio Kojima, Shunichi Nozawa, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Bipedal walking control against swing foot collision using swing foot trajectory regeneration and impact mitigation. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4531–4537. IEEE, 2017.

[21] ORBBEC. ASTRA. `https://orbbec3d.com/product-astra/`.