Global Navigation for Humanoid Robots Using Sampling-Based Footstep Planners

Zeyang Xia, *Member, IEEE*, Jing Xiong, and Ken Chen, *Member, IEEE*

***Abstract*—The objective of humanoid navigation is to plan the global locomotion path for humanoid robots in complex global environments. It is different from wheeled mobile robot navigation because of the biped's unique abilities to step upon/over obstacles and the stability requirement is difficult. Sampling-based footstep planning that considers the aforementioned characteristics is a goal-directed navigation approach for humanoid robots. This paper will describe two footstep planning approaches, including our improvements on the previous deterministic sampling-based approach and the newly proposed randomized sampling-based approach. Numerical experiments are also given to verify the feasibility and planning performance of the two approaches.**

***Index Terms*—Biped navigation, deterministic sampling, footstep planner, randomized sampling, sampling.**

I. Introduction

In the past 15 years, we have witnessed rapid progress in research on biped humanoid robots. As we position humanoid robots as future agents serving humans in their daily lives, as well as executing tasks in certain environments instead of humans, advanced locomotion capabilities in existing human-living environments are demanded for humanoid robots. Two topics are related to this study: biped locomotion pattern and global navigation.

Regarding the environmental information considered in the planning, most existing biped locomotion patterns can be categorized into two classes. The first class aims to realize stable locomotion in simple, structured, obstacle-free environments [1]–[6], such as walking on flat floors, slopes, or staircases. In this class, the robot does not need to perceive the environment, or only certain environment-related locomotion pattern parameters are required. The second class aims to realize collision-free locomotion in structured environments with obstacles. The robot needs to perceive the local obstacle information by sensors (stereovision system, lasers, etc.) [7], [8]. These two previous classes are essentially locomotion patterns in local environments. However, to accomplish tasks in human-living environments where complex obstacles with various forms exist, such as homes and offices, humanoid robots must bear the ability of global navigation in wide areas.

A number of approaches for mobile robot navigation have been studied and implemented [9], and these approaches usually require the robot to circumvent obstacles. In contrast, humanoid robots have the ability to step over/upon certain obstacles. Due to this unique locomotion ability, it is difficult to predefine the objects in the environment as "obstacle" or "nonobstacle."

Due to these difficulties, the previous navigation approaches for mobile robots are not practical for accomplishing the humanoid navigation.

Kuffner *et al.* [10]–[12] proposed an algorithm for global footstep planning in obstacle-clustered environments in which the planner needs no preprocessing of the environmental information, and collision is avoided by collision checkers. This algorithm is essentially a sampling-based approach, and other studies based on this algorithm have also been conducted. Ayaz *et al*. [13] improved the smoothness of the trajectories for posture transitions; Michel *et al.* [14] applied this approach in some dynamic environments; and Chestnutt *et al.* applied the dynamically adjustable action model [15] and the adaptive action model to perform local searches based on reference actions [16].

In order to improve the planning efficiency and feasibility of the footstep planner for biped navigation (both in common 3-D environments and some other special environments, such as fields with local minima or narrow passages), we accomplished two works in our study, which are as follows: 1) improving the previous deterministic sampling-based approach by using dynamically selected footstep-transition models, and 2) applying a randomized sampling strategy with goal-biased control to the footstep planner. Our study concentrates on the realization and implementation of the newly proposed sampling strategies employed in the footstep planner. And the frame of footstep planners we used is based on the planner given in [10]. The implementations discussed in this paper are realized in static environments. Considering the excellent performance of their planning efficiency, the proposed planners can be adopted in planners for dynamic environments, which will allow the robot to accomplish the step-by-step replanning based on simultaneously acquired environmental information.

The remainder of this paper is organized as follows. Section II introduces the basic sampling-based footstep planner. Sections III and IV describe the deterministic approach using the dynamic footstep-transition model, the randomized approach, and their implementations, respectively. Section V concludes the paper.

Manuscript received August 22, 2009; revised January 24, 2010 and May 10, 2010; accepted May 18, 2010. Date of publication July 1, 2010; date of current version May 11, 2011. Recommended by Technical Editor M. O'Malley.

Z. Xia was with the Department of Precision Instruments and Mechanology, Tsinghua University, Beijing 100084, China. He is now with the Department of Mechanical Engineering, Indiana University-Purdue University, Indianapolis, IN 46202-5160 USA (e-mail: zeyang.xia@ieee.org).

J. Xiong was with the Department of Precision Instruments and Mechanology, Tsinghua University, Beijing 100084, China. She is now with the Department of Biomedical Engineering, Indiana University-Purdue University, Indianapolis, IN 46202-5160 USA.

K. Chen is with the Institute of Manufacturing Engineering, Department of Precision Instruments and Mechanology, Tsinghua University, Beijing 100084, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMECH.2010.2051679

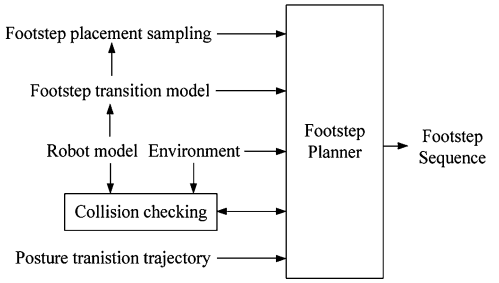

Fig. 1. Sampling-based footstep planner for humanoid robots.

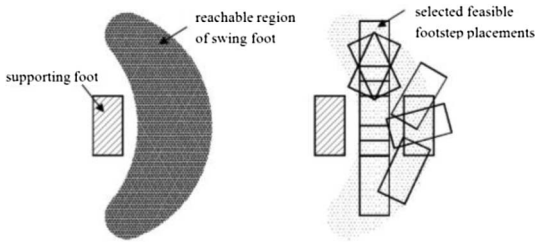

Fig. 2. Footstep-transition model. (Left) Region reachable for the swing foot, which expands to a 3-D space while height offsets are considered. (Right) Set of foot placements defined in a footstep-transition model.

II. Sampling-Based Footstep Planner

The sampling-based footstep planner first builds a search tree originated from the starting footstep placement of the robot. The search tree expands by footstep-placement sampling in the planning space. Footstep placements that cause collisions are pruned from the tree by the collision checker based on the state of the robot and environmental information. The planning completes when one footstep placement in the search tree reaches the goal region or fails once the planner reaches planning limitation (duration or sampling steps). Fig. 1 gives a block diagram of the footstep planner.

The footstep-placement sampling that dominates the search-tree expansion distinctly affects the efficiency and completeness of the planner. The footstep placements are sampled in the planning space according to the footstep-transition model, which is denoted as $\boldsymbol{\mathcal{T}}$, in which a set of discrete feasible foot placements is defined for each swing foot (see Fig. 2). The construction of the footstep-transition model must include consideration of the following factors: 1) the physical kinodynamics constraints, which require selected footstep placements to be located within the reachable space of the swing foot considering the kinematics and dynamic constraints of the physical robot; 2) functionalities, which require selected footstep placements to be optimally grouped to perform different locomotion functions, such as walking forward and backward, turning, side movement, etc.; and 3) the stability criterion, which requires that the physical robot can walk, as defined by the footstep-transition model with stable gesture-transition trajectories.

The gesture-transition trajectory is defined as the trajectory with which the robot transfers support of its gesture from the anterior foot placement to the posterior foot placement. The gesture transition is performed by using two intermediate symmetric statically stable gestures, a method previously proposed in [11]. Based on this method, we developed an approach to posture-transition-trajectory planning with low online computational capacities, which includes: 1) an "offline generation–online calling" mode for gesture transition without height offset between adjacent footstep placements, and 2) an "offline generation–online modification" mode for transitions with height offset. The zero-moment-point (ZMP) stability criterion is used in the posture-transition-trajectory planning to ensure the locomotion stability [17]. The concept and realization of the posture-transition-trajectory planning are discussed in detail in [17] and [19].

Collision checking is a critical issue affecting the computational complexity of planning. Kuffner *et al.* [10] proposed a two-leveled gradient collision-checking algorithm, including: 1) 2-D collision checking between obstacles and target foot placement and 2) 3-D collision checking between obstacles and the robot, while it transfers to the target posture. Based on this algorithm, we proposed a fast collision-checking algorithm with lower complexity to avoid full 3-D checking. This algorithm is realized by interpolating virtual foot placements for sole-obstacle checking during the robot's posture transition. A prerequisite for this algorithm is that the robot is not supposed to put its foot under an obstacle (such as a chair), even if it will not cause a collision. With this prerequisite, the robot would not encounter a 3-D collision as long as the interpolated foot placements would not encounter a 2-D collision with a reasonable safety margin during its posture transition. Considering the planning environments, this prerequisite is quite acceptable. The proposed algorithm provides satisfying computational performance. Details of the collision checking are available in [17].

The physical robot model, collision checking algorithms, gesture-transition-trajectory planning, and stability criterion involved in the footstep planner design in our study are based on our previously developed THBIP-2 humanoid robot, which is 70 cm in height and 18 kg in weight with 24 DOFs [18].

III. Deterministic Sampling-Based Footstep Planner

A. Dynamic Footstep-Transition Model

Existing footstep planners use deterministic sampling strategies [10]–[16]. Sufficient feasible foot placements are required in $\boldsymbol{\mathcal{T}}$ to ensure the planning completeness. However, the number of foot placements in $\boldsymbol{\mathcal{T}}$, which is denoted as $N_{\mathcal{F}}$, decides the branching factor of the search tree. The branching factor is the number of children at each node of a search tree, and it decides the complexity of works following each node. A large $N_{\mathcal{F}}$ will increase the search complexity remarkably. In order to achieve an optimization considering the planning efficiency, which requires $\boldsymbol{\mathcal{T}}$ with a small $N_{\mathcal{F}}$, and the planning completeness, which requires $\boldsymbol{\mathcal{T}}$ with a large $N_{\mathcal{F}}$, we propose a dynamic footstep-transition model, which is denoted as $\boldsymbol{\mathcal{T}}_{\mathrm{dyn}}$. This method is brought forward on two hypotheses based on numerical planning experiments.

Hypothesis I: During the footstep planning in nonextreme environments, the foot placements performing the basic

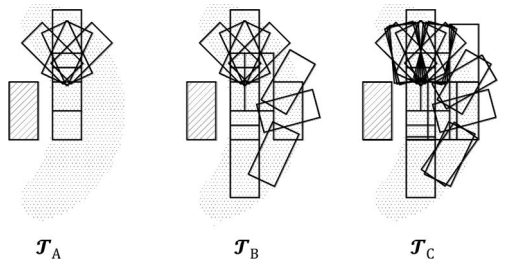

Fig. 3. Dynamic footstep-transition model denoted as $\mathcal{T}_{\rm dyn}$ is composed of three submodels with increasing $N_{\mathcal{F}}$. We have $\mathcal{T}_A \subset \mathcal{T}_B \subset \mathcal{T}_C$.

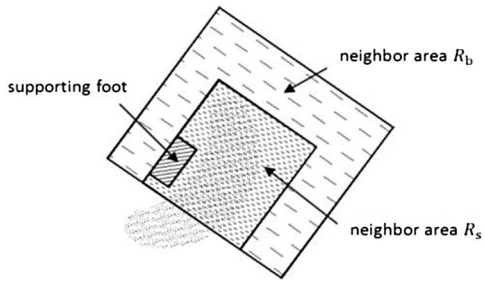

Fig. 4. Neighboring areas denoted as R_s and R_b. Dimensions of R_s and R_b are not given in the real scale.

locomotion functions in $\mathcal{T}$ are called with higher probabilities than those performing nonbasic functions.

Hypothesis II: The more complex the environments are, the higher the probability that the foot placements performing nonbasic locomotion functions will be called.

The dynamic footstep-transition model is composed of three submodels with increasing $N_{\mathcal{F}}$ (see Fig. 3). The sets of foot placements of anterior submodels are true subsets of posterior ones. And the submodels are called at each sampling step according to the assessment of neighboring areas of the current supporting foot (see Fig. 4), and the dimensions of these areas are assigned, considering how far the robot may reach with the next step or the next three steps if it walks forward or turns slightly. The assessment is processed with two indices, a representing height index and a height-variance index. Compared to [16], the proposed model accomplishes its adaptation to the environment in a wider area, not aiming at specific obstacles, and the branching factor may vary at each search step. Fig. 5 gives the submodels' calling flow.

Definition I: Representing height, which is denoted as $h_{\rm rep}$, represents the height of a nonflat area. $h_{\rm rep}$ of area A is given as follows:

$$h_{\rm rep}(A) = h(A_{\rm rep}) \tag{1}$$

where $\text{area}(A_{\rm rep}) = \max\{\text{area}(A_1), \text{area}(A_2), \ldots, \text{area}(A_n)\}$, $\{A_1, A_2, \ldots, A_n\} = A$ is a partition of A according to the height of the subareas, and $\text{area}(A_k)$ computes the area of subarea A_k.

Definition II: Height variance, which is denoted as $I_{\rm var}$, represents the height variance relative to $h_{\rm rep}$ of the assessed area.

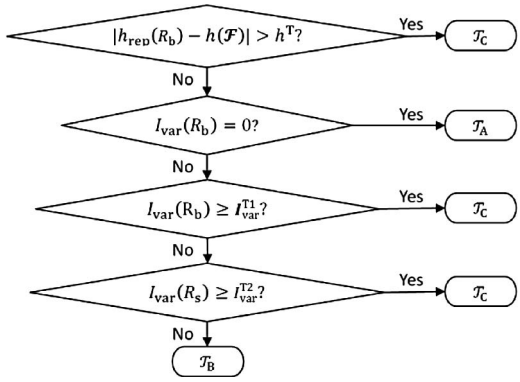

Fig. 5. Submodels' calling flow. $I_{\rm var}^{T1}$ and $I_{\rm var}^{T2}$ are height-variance thresholds. h^T is the threshold of the representing height difference between the assessed area R_b and the supporting foot. $h(\mathcal{F})$ is the height of the current supporting foot.

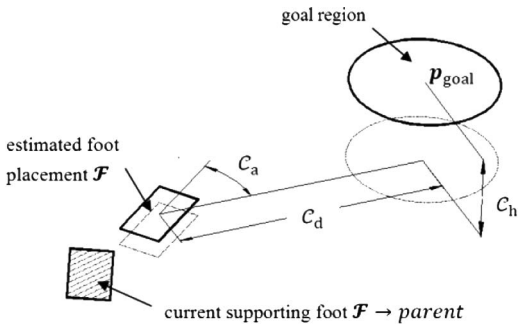

Fig. 6. Subitems of the heuristic function for sampling directing.

$I_{\rm var}$ is given as follows:

$$I_{\rm var} = \sqrt{\frac{1}{N-1}\sum_{\boldsymbol{g}\in A}(h_{\boldsymbol{g}} - h_{\rm rep})^2} \tag{2}$$

where $h_{\boldsymbol{g}}$ is the height of grid $\boldsymbol{g}$ in area A and N is the number of grids in area A.

B. Realization

A function $\boldsymbol{H}(\mathcal{F})$ computing the cost of a target foot placement is configured by considering its inherent priority in $\mathcal{T}$ and weighted heuristic items, including its distance, height offset, and orientation to the goal region $\boldsymbol{p}_{\rm goal}$(see Fig. 6)

$$\boldsymbol{H}(\mathcal{F}) = k_d \mathcal{C}_d(\mathcal{F}, p_{\rm goal}) + k_a \mathcal{C}_a(\mathcal{F}, p_{\rm goal}) + k_h \mathcal{C}_h(\mathcal{F}, p_{\rm goal}) \tag{3}$$

where k_d, k_a, and k_h are the weights. Similar functions were also given in [10] and [11].

We realize the deterministic sampling-based footstep planning with A* algorithm (see Fig. 7).

C. Implementation

We implemented the deterministic sampling-based footstep planner using dynamic footstep-transition models in multitype environments. Figs. 8–11 show some generated footstep sequences in various environments and Table I gives corresponding data. The planning results verify the good

Deterministic footstep planning using A-star

1 Initial Footprint List FPlist, configure cost function $\mathcal{C}(\mathcal{F})$
2 compute cost of $\mathcal{F}_{\text{init}}$, add $\mathcal{F}_{\text{init}}$ to FPlist ;
3 **repeat while**(sampling time < planning duration upper limit),
4 get head node of FPlist $\mathcal{F}_{\text{head}}$
5 assess neighbor areas of $\mathcal{F}_{\text{head}}$ and call the submodel of $\mathcal{T}_{\text{dyn}}$
6 extend the child footprint of $\mathcal{F}_{\text{head}}$, $\mathcal{F}_{\text{head}} \rightarrow child[i]$
7 collision checking and mark the footprints collide
8 **if** $\mathcal{F}_{\text{head}} \rightarrow child[i] \in \boldsymbol{F}_{goal}$, jump to 12
9 **otherwise**
10 compute costs of rest child nodes $\mathcal{F}_{\text{head}} \rightarrow child[i]$
11 and insert them to FPlist, ensuring the head node of PFlist is with the smallest cost
12 search succeeds,output sequence from $\mathcal{F}_{\text{init}}$ to $\mathcal{F}_{\text{head}} \rightarrow child[i]$ in 8
13 search fails in required computing time or sampling time

Fig. 7. Deterministic footstep planner using A*. $\boldsymbol{F}_{\text{goal}}$ is the goal region.

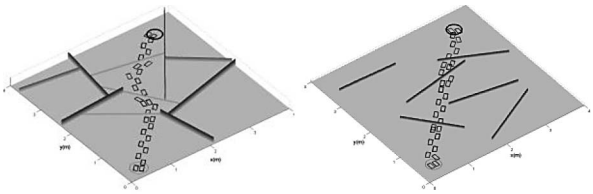

Fig. 8. Footstep sequences in environments with (a) height-varying strip obstacles and (b) low strip obstacles. The light and dark circles denote the starting and goal region separately, and the different colors on the top surfaces denote varying heights.

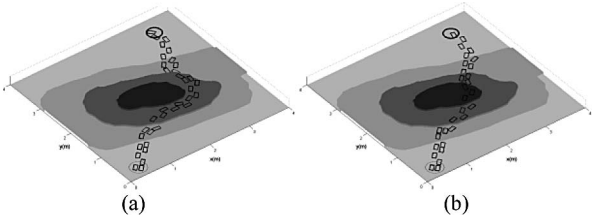

Fig. 9. Footstep sequences in the same environments with (a) high and (b) low height difference penalties in the heuristic function.

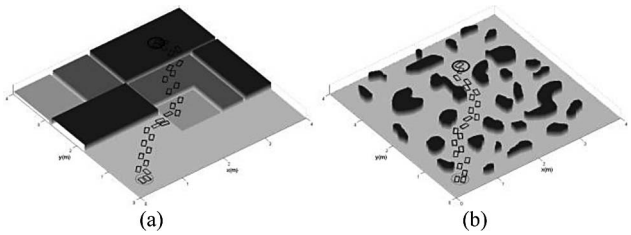

Fig. 10. Footstep sequences in environments with staircases with (a) grooves and (b) high obstacles.

performance of both efficiency and completeness of the deterministic sampling-based footstep planner using dynamic footstep-transition models.

In addition, we compare the deterministic footstep planners with dynamic footstep-transition models, i.e., $\mathcal{T}_{\text{dyn}}$ and single footstep-transition models, i.e, $\mathcal{T}_A$, $\mathcal{T}_B$, and $\mathcal{T}_C$ included in $\mathcal{T}_{\text{dyn}}$. The comparison planning is accomplished in environments, such as: 1) clear flat area; 2) in Fig. 8(b); 3) in Fig. 10(a); and 4) in Fig. 10(b). Fig. 12 compares nodes searched and Table II gives the corresponding parameters used. From the results, we can see that planning with $\mathcal{T}_{\text{dyn}}$ finishes with searched nodes, which are less than those planned with single $\mathcal{T}$, or at least as low as the best of those planned with single $\mathcal{T}$. The essence of dynamic footstep-transition models is to increase feasibility and efficiency of the footstep planner by using varying branching factors adaptive to the environment.

According to our simulation experiences, the planner using $\mathcal{T}_{\text{dyn}}$ is not very sensitive to these parameters. For the weights in the cost function, we are using empirical sets of values for different types of environments, but these values are slightly different from each other. Even when the same set of parameters is used for different environments, the planning results may vary, but the quality of the results will not be affected. Also, parameters can be used to realize different locomotion "habits." In Fig. 9, we use different height penalties controlled by k_h. For details, see [19].

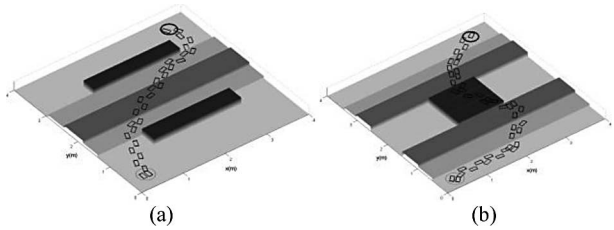

Fig. 11. Footstep sequences in environments with different staircases.

TABLE I
DATA OF PLANNING USING DETERMINISTIC FOOTSTEP PLANNER*

Item	N_{nod}	$t(s)$	N_{step}
Fig.8 (a)	673	0.068	27
Fig.8 (b)	705	0.076	26
Fig.9 (a)	1893	0.182	35
Fig.9 (b)	1355	0.143	28
Fig.10 (a)	1427	0.151	27
Fig.10 (b)	545	0.059	20
Fig.11 (a)	1651	0.177	28
Fig.11 (b)	4287	0.433	41

*N_{nod} is the number of nodes searched in the planning. It is an important index to assess search algorithms, which is independent of computing hardware, different from planning duration, denoted by t. N_{step} is the number of steps of generated footstep sequences.

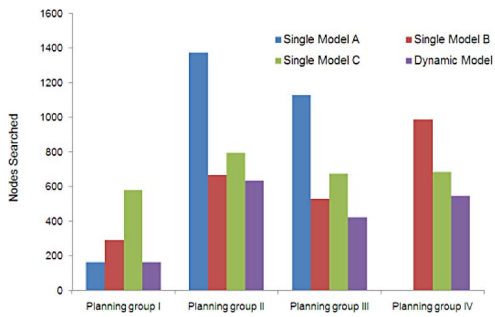

Fig. 12. Comparison of nodes searched using different $\mathcal{T}$. Note that planning using $\mathcal{T}_A$ with few foot placements fails in IV.

TABLE II
PARAMETERS FOR COMPARISON PLANNING*

Parameter	I	II	III	IV
k_{d}	1	1	1	1
k_{a}	0.04	0.025	0.01	0.01
k_{i}	0.02	0.01	0.01	0.01
$k_{\mathrm{ph}}/k_{\mathrm{h}}$	0.05	0.05	0.05	0.05
k_{s}	0.02	0.005	0.005	0.005
$\mathrm{k}_{\mathrm{exp}}$	1.05	1.1	1.1	1.1
h^T		0.2 m		
l_{var}^{T1}		0.15 m		
l_{var}^{T2}		0.07 m		

*k_i is the weight of the distance from the assessed foot placement to the starting point in the cost function. k_{ph} is the weight of the height offset between the assessed foot placement and the current supporting placement. k_s is the weight of the self-character of the assessed foot placement.

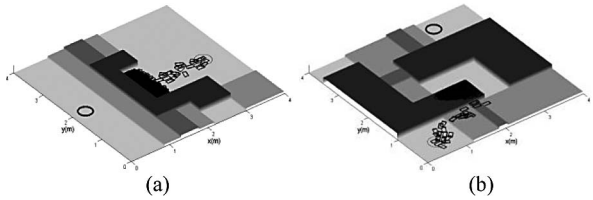

Fig. 13. Two cases of search trees trapped in local loops and dead ends with planners based on A* algorithms.

IV. RANDOMIZED SAMPLING-BASED FOOTSTEP PLANNER

In this section, we propose a randomized sampling-based footstep planner based on rapidly exploring random trees (RRT) algorithm, multi-RRT footstep planner, which is capable of resolving planning problems in "unfriendly" environments, such as areas with local minima or/and narrow passages.

A. Problem Description and Randomized Sampling Strategy

Previous approaches, such as the one in Section III, are realized based on deterministic sampling strategies, in which the search-tree expansion is mainly directed by deterministic functions. These functions are designed aiming to generate relatively optimal sequences of footstep placements, such as smooth sequences with the least number of footsteps and least unnecessary footsteps to step over/upon obstacles. The deterministic sampling-based approaches are very practical in open-area environments. However, a critical problem is that they rely on the sampling set design of the footstep-transition model and sampling-directing functions. This characteristic may cause planning failures: the search tree may not be able to converge, i.e., get trapped, in required planning duration or sampling steps. This situation is especially apparent during the implementation in areas with local minima or/and narrow passages, which exist in human-living environments, such as homes or office buildings. Fig. 13 shows two cases of the trapped search trees (see [12] for more cases).

Besides the improvement on the sampling-directing functions, another option regarding this problem is to improve the ability of traversing in local loops and dead ends of the search tree by employing randomized sampling strategies. We apply the random sampling strategy to the humanoid footstep planning in our study. The proposed random sampling strategy is based on RRT [20].

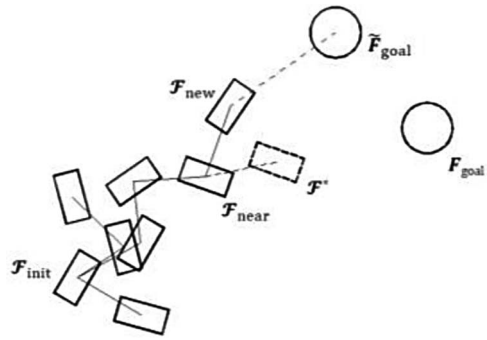

Fig. 14. Single-step expansion of the search tree of the RRT-based footstep planner. $\mathcal{F}_{\mathrm{init}}$ denotes the initial foot placement.

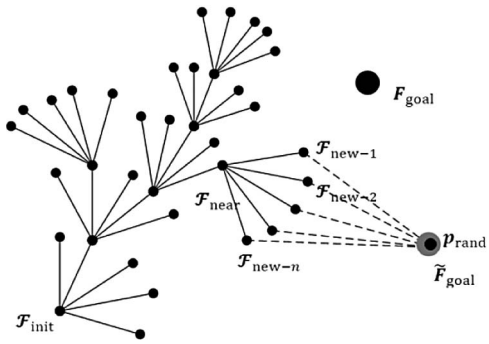

Fig. 15. Single-step expansion of the search tree of the multi-RRT footstep planner. Each point denotes one foot placement.

The key issue of the RRT-based footstep planner is to produce a temporary goal region randomly distributed in the planning space at each planning step. The search-tree expands are directed by these temporary goal regions, instead of the actual goal. This mechanism provides the search tree with an improved ability to avoid obstacles and traversing local loops or dead ends. Fig. 14 explains three procedures of a single-step expansion of the search tree: 1) to produce a temporary goal region $\tilde{\boldsymbol{F}}_{\mathrm{goal}}$; 2) to find the foot placement in the search tree nearest to $\tilde{\boldsymbol{F}}_{\mathrm{goal}}$ $\boldsymbol{\mathcal{F}}_{\mathrm{near}}$; and 3) to add a new foot placement $\boldsymbol{\mathcal{F}}_{\mathrm{new}}$ to the search tree.

B. Multi-RRT Footstep Planner

In Fig. 14, we can see that the basic RRT-based footstep planner only adds one foot placement $\boldsymbol{\mathcal{F}}_{\mathrm{new}}$ to the search tree. $\boldsymbol{\mathcal{F}}_{\mathrm{new}}$ is selected regarding its distance to $\tilde{\boldsymbol{F}}_{\mathrm{goal}}$, while $\boldsymbol{\mathcal{F}}^*$, which has a smaller distance to $\boldsymbol{F}_{\mathrm{goal}}$, is not added. The previous characteristic that RRT only adds one node to the search tree may result in ill-conditioned footstep sequences. In order to resolve this problem, we modify the RRT-based footstep planner by adding all the foot placements in the footstep-transition model to the search tree at each step. We term this planner as *multi-RRT footstep planner*. Fig. 15 shows the single-step search tree of the multi-RRT-based footstep planner, and Fig. 16 gives the

Multi-RRT algorithm

1 Initial random tree T, add $\mathcal{F}_{\text{init}}$ to T;
2 **repeat while**(sampling time < required largest sampling time),
3 produce a random point $\boldsymbol{p}_{\text{rand}}$ and create $\tilde{\boldsymbol{F}}_{\text{goal}} = (\boldsymbol{p}_{\text{rand}}, d_{\min})$;
4 find $\mathcal{F}_{\text{near}}$ nearest to $\tilde{\boldsymbol{F}}_{\text{goal}}$;
5 **if** $\rho_2(\mathcal{F}_{\text{near}}, \boldsymbol{p}_{\text{rand}}) < d_{\min}$, jump to 3
6 **otherwise** $\mathcal{F}_{\text{new}-i}(\text{i} = 0,1,\ldots,\text{n}-1) \leftarrow \text{Extend}(\mathcal{F}_{\text{near}})$
7 **if** $\mathcal{F}_{\text{new}-i} \in \boldsymbol{F}_{\text{goal}}$, **jump** to 9
8 **otherwise** add $\mathcal{F}_{\text{new}-i}$ to T
9 search succeeds, output sequence from $\mathcal{F}_{\text{init}}$ to $\mathcal{F}_{\text{new}-i}$ in 7
10 search fails in required computing time or sampling time

Fig. 16. Algorithm of the multi-RRT footstep planner. ρ_2 denotes the measure function from a foot placement to a region. It considers not only the Euclidean distance, but also the height offset, orientation, etc. The dimension of the temporary goal region is denoted by $d_{\min}$, i.e., the diameter of the circle denoting $\tilde{\boldsymbol{F}}_{\text{goal}}$, as shown in Fig. 14.

Create Temporary Goal Region of Multi-RRT-GoalBias

1 Produce a probability number $P = \text{rand}() \in [0, 1]$
2 **if** $P < P_{\text{goal}}$
3 return actual goal $\boldsymbol{F}_{\text{goal}}$ as temporary goal region $\tilde{\boldsymbol{F}}_{\text{goal}}$
4 **otherwise**
5 Return randomized $\tilde{\boldsymbol{F}}_{\text{goal}} = (\boldsymbol{p}_{\text{rand}}, d_{\min})$

Fig. 17. Goal-biased random sampling controlled by P_{goal}. $P = \text{rand}()$ is a computer-generated probability number uniformly distributed in [0,1].

multi-RRT algorithm. Multi-RRT inherits the ability of quick expansion in the planning space from the RRT algorithm. The more the nodes are added to the search tree in each step, the quicker the search tree covers the planning space. Numerical results also show that the search tree of multi-RRT may cover the planning space uniformly after sufficient sampling times, which is similar to the characteristic of probabilistically completeness of the basic RRT algorithm.

C. Goal-Biased Control

According to the numerical sampling experiments as well as the theoretical analysis, the randomly sampled foot placements of the multi-RRT footstep planner distribute uniformly in the planning space after sufficient sampling steps. In order to improve the rate that the random tree converges to the goal region, we apply a nonuniform control to the sampling: The sampling distribution is biased to the goal region controlled by a parameter termed as goal-probability threshold (GPT, denoted as $P_{\text{goal}} \in [0, 1]$). When producing $\tilde{\boldsymbol{F}}_{\text{goal}}$ (see step 3 in Fig. 16), the planner returns $\boldsymbol{F}_{\text{goal}}$, instead $\tilde{\boldsymbol{F}}_{\text{goal}}$, at a probability of P_{goal} (see Fig. 17). We termed this approach as the *multi-RRT-GoalBias footstep planner*.

P_{goal} is a predefined parameter that controls the expansion and convergence of the random tree. Sampling experiments controlled by varying P_{goal} values show that [22]: 1) P_{goal} continuously affects the convergence rate of random trees, and 2) the random trees quickly converge to the goal with low sampling steps, if P_{goal} is set with a proper value. These observations verify that an optimized P_{goal} exists, with which the random trees converge at the quickest rate. The optimized P_{goal} value is decided by the planning problem. For each type of planning problem, nonlinear mapping from the parameters describing the planning problem to optimized P_{goal} can be fitted so that the optimized P_{goal} value can be obtained. We use a back propagation (BP) neural network to fit the mapping, which is specifically discussed in [22].

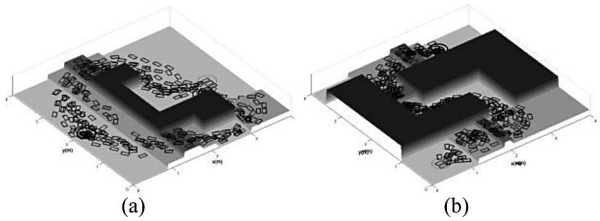

Fig. 18. Superimposed footstep sequences by multi-RRT-GoalBias footstep planners in environments with (a) local minima and (b) narrow passages.

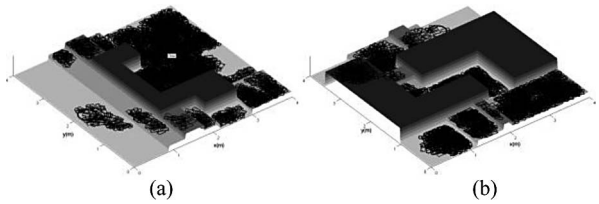

Fig. 19. Search-tree expansions by the multi-RRT-GoalBias footstep planner.

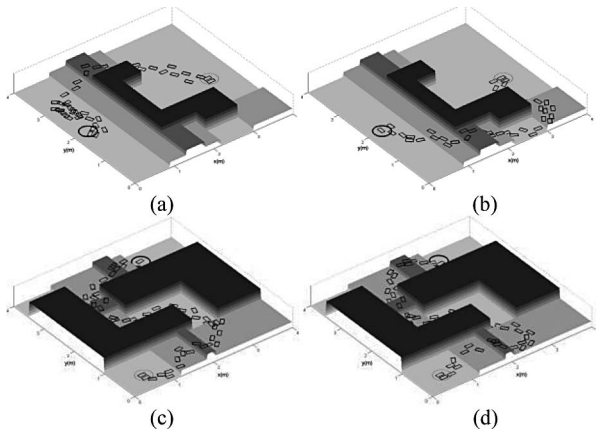

Fig. 20. Ill-conditioned footstep sequences generated by the basic RRT-based footstep planner.

D. Implementation and Optimization

We implement the multi-RRT-GoalBias footstep planer in environments with local minima, narrow passages, or other unfriendly environments. For each planning problem, 200 trials of randomized planning were conducted. Fig. 18 shows ten superimposed footstep sequences generated by the multi-RRT-GoalBias footstep planner in environments with local minima and narrow passages, respectively (both are randomly selected trials from the 200 trials). Fig. 19 shows expansion of the search trees by the multi-RRT-GoalBias footstep planner, compared to the search trees that were trapped by deterministic approaches (see Fig. 13). Fig. 20 shows typical ill-conditioned footstep sequences generated by basic RRT-based footstep planners, which

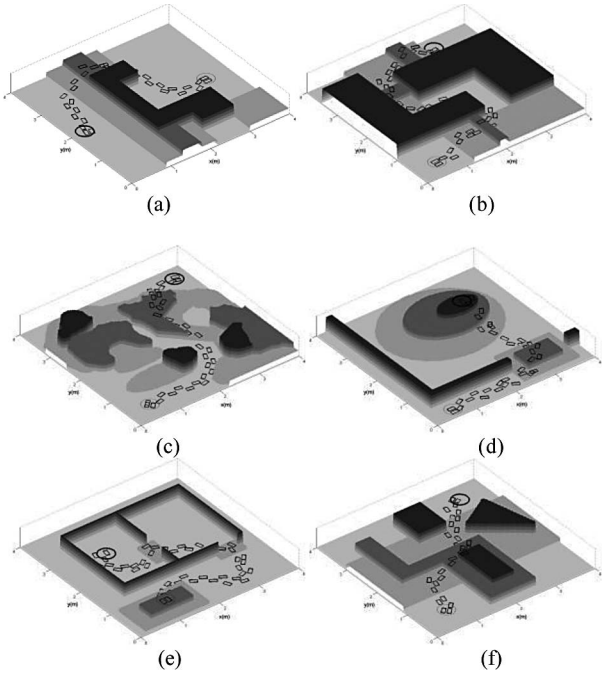

Fig. 21. Footstep sequences after pseudooptimization by the multi-RRT-GoalBias footstep planner.

only extend one foot placement at each sampling step (see Section IV-B), in which undesired turning, side steps, or unnecessary step adjustments exist. These comparisons verify that 1) the search trees by the multi-RRT-GoalBias footstep planner bear notable ability to traverse local loops and dead ends, and 2) the multi-RRT-GoalBias footstep planner can avoid ill-conditioned results, which basic RRT-based planners may generate.

We currently use a pseudooptimization method in the planner. This method constructs a quantitative function to assess generated footstep sequences, when considering the number of footsteps, smoothness, and frequencies of significant height and orientation changes, which is related to the risk of stability lose. According to our planning experiments, an acceptable footstep sequence can be obtained out of each planning set with very few trials, ranging from two to five (see Fig. 18 and note that there are ten footstep sequences superimposed in each figure). In implementations with firm planning duration requirements, this pseudooptimization can be used to avoid unacceptable results after two trials. In addition, the design of the algorithm to find $\mathcal{F}_{\mathrm{near}}$ (see step 4 in Fig. 16) also helps to smooth the results. Planning results after pseudooptimizations are given in Fig. 21 (all results are based on two random trials). Compared to the results generated by basic-RRT planners shown in Fig. 20, the footstep sequences by the multi-RRT-GoalBias footstep planer are better. Data of 200 trials of planning in environments shown in Figs. 20 and 21 are given in Table III. Less than five trials out of 200 trials of planning in Fig. 21 may fail, although they are set with a very small upper limit of sampling steps as 1×10^3.

Particularly, Figs. 13(a), 20(a) and (b), and 21(a) give results in the same environment with local minima by the A*-based planner, the basic RRT-based planner, and the multi-RRT-GoalBias planner, respectively. Figs. 13(b), 20(c) and (d), and 21(b) give results in the same environment with narrow passages by the A*-based planner, the basic RRT-based planner, and the multi-RRT-GoalBias planner, respectively. The comparisons of planning results in same environments verify the good performance of the multi-RRT-GoalBias planner.

TABLE III
STATISTICAL DATA OF 200 TRIALS USING MULTI-RRT-GOALBIAS*

Item	P_{goal}	$\bar{N}_{\mathrm{sam}}$	$\bar{N}_{\mathrm{nod}}$	$\bar{t}\ (s)$	$\bar{N}_{\mathrm{step}}$
Fig.20(a)/(b)	-	8119	8122	0.835	43
Fig.20(c)/(d)	-	14413	14421	1.469	58
Fig.21(a)	0.151	489	4898	0.477	31
Fig.21(b)	0.109	825	8255	0.818	44
Fig.21(c)	0.381	297	2979	0.304	34
Fig.21(d)	0.325	253	2537	0.253	34
Fig.21(e)	0.168	381	3813	0.373	45
Fig.21(f)	0.398	274	2748	0.261	28

*$\bar{N}_{\mathrm{sam}}$ are the average sampling steps, $\bar{N}_{\mathrm{nod}}$ is the average number of searched nodes, and $\bar{t}$ is the average planning duration. The statistical process is described in [24].

Drawbacks of the current pseudooptimization method include two aspects, which are as follows: 1) the computational capacity increases since additional trials need to be conducted and 2) the results after the pseudooptimization are not theoretically ensured to be the most optimized results, although ill-conditioned results can be avoided. What is considered in our ongoing development of potential full optimization methods is that the full optimization of footstep sequences would not cause additional online posture-transition-trajectory planning or additional collision checking, which increase the computational capacity of planning.

V. DISCUSSION AND CONCLUSION

We present our progress in footstep planning for global navigation of humanoid robots, employing deterministic and randomized sampling strategies. Our works include: 1) improving previous deterministic sampling-based approaches by using dynamic footstep-transition models, and 2) realizing a randomized sampling-based footstep planner with a goal-biased control strategy. To the best of our knowledge, our study is the first attempt to apply random sampling strategies for humanoid footstep planning.

We use the plain A* algorithm to realize the deterministic footstep planner and a modified RRT algorithm to realize the randomized footstep planner in this paper. The planners in this paper are termed only regarding to strategies used to extend samples. Actually, it is also possible to use more complicated algorithms to design the planner, such as A* with randomized sampling strategies and RRT with deterministic sampling strategies.

Although the randomized sampling-based footstep planner has been implemented with desirable results, some features of the randomized approach are still based on the numerical experiments. The mathematical proofs and full-optimization of the planning results will be the focuses of our future works. We are

also in the process of collaborating with other teams on global map construction to implement the proposed methods on our physical robot.

ACKNOWLEDGMENT

The authors would like to thank Dr. J. J. Kuffner at the Robotics Institute, Carnegie Mellon University, for his helpful advice during the early stages of our study, and the anonymous reviewers for their thorough and instructional comments, which not only improved this paper, but also provided guidance for our future work.

REFERENCES

[1] E. T. Esfahani and M. H. Elahinia, "Stable walking pattern for an SMA actuated biped," *IEEE/ASME Trans. Mechatronics*, vol. 12, no. 5, pp. 534–541, Oct. 2007.

[2] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Yokoi, and H. Hirukawa, "A realtime pattern generator for biped walking," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, 2002, pp. 31–37.

[3] G. Endo, J. Morimoto, T. Matsubara, J. Nakanish, and G. Cheng, "Learning CPG-based biped locomotion with a policy gradient method: Application to a humanoid robot," *Int. J. Robot. Res., Spec. Issue Mach. Learn. Robot.*, vol. 27, no. 2, pp. 213–228, 2008.

[4] V. Sangwan and S. K. Agrawal, "Differentially flat design of bipeds ensuring limit cycles," *IEEE/ASME Trans. Mechatronics*, vol. 14, no. 6, pp. 647–657, Dec. 2009.

[5] T. Aoyama, Y. Hasegawa, K. Sekiyama, and T. Fukuda, "Stabilizing and direction control of efficient 3-D biped walking based on PDAC," *IEEE/ASME Trans. Mechatronics*, vol. 14, no. 6, pp. 712–718, Dec. 2009.

[6] T. Geng and J. Q. Gan, "Planar biped walking with an equilibrium point controller and state machines," *IEEE/ASME Trans. Mechatronics*, vol. 15, no. 2, pp. 253–260, Apr. 2010.

[7] R. Cupec, O. Lorch, and G. Schmidt, "Vision-guided humanoid walking—Concepts and experiments," presented at the 12th Int. Workshop Robot. Alpe-Adria-Danube Region, Cassino, Italy, May 2003.

[8] M. Yagi and V. J. Lumelsky, "Biped robot locomotion in scenes with unknown obstacles," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Detroit, MI, May 1999, pp. 375–380.

[9] J. C. Latombe, *Robot Motion Planning*. Boston, MA: Kluwer, 1991.

[10] J. J. Kuffner, K. Nishiwaki, K. Kagami, M. Inaba, and H. Inoue, "Footstep planning among obstacles for biped robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Maui, Hawaii, Oct. 2001, pp. 500–505.

[11] J. J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba, and H. Inoue, "Motion planning for humanoid robots," *Trans. Adv. Robot.*, vol. 15, pp. 365–374, 2005.

[12] J. Chestnutt, J. J. Kuffner, K. Nishiwaki, and S. Kagami, "Planning biped navigation strategies in complex environments," presented at the IEEE Int. Conf. Humanoid Robot., Munich, Germany, 2003.

[13] Y. Ayaz, K. Munawar, M. B. Malik, A. Konno, and M. Uchiyama, "Human-like approach to footstep planning among obstacles for humanoid robots," *Int. J. Humanoid Robot.*, vol. 4, no. 1, pp. 125–149, 2007.

[14] P. Michel, J. Chestnutt, J. Kuffner, and T. Kanada, "Vision-guided humanoid footstep planning for dynamic environments," in *Proc. IEEE/RAS Int. Conf. Humanoid Robots*, Tsukuba, Japan, 2005, pp. 13–18.

[15] J. Chestnutt, M. Lau, J. Kuffner, J. Hodgins, and T. Kanada, "Footstep planning for the Honda ASIMO humanoid," in *Proc. IEEE Int. Conf. Robot. Autom.*, Tsukuba, Japan, 2005, pp. 629–634.

[16] J. Chestnutt, K. Nishiwaka, J. Kuffner, and S. Kagami, "An adaptive action model for legged navigation planning," in *Proc. IEEE/RAS Int. Conf. Humanoid Robots*, 2007, pp. 196–202.

[17] Z. Y. Xia, "Sampling-based footstep planning for humanoid robots," Ph.D. thesis, Tsinghua Univ., Beijing, China, 2008.

[18] Z. Xia, L. Liu, K. Chen, Q. Yi, and J. Xiong, "Design aspects and developments of humanoid robot THBIP-2," *Robotica*, vol. 26, no. 1, pp. 109–116, 2008.

[19] Z. Xia, J. Xiong, and K. Chen, "A deterministic sampling-based approach to global footstep planning for humanoid robots," in *Proc. 2009 IEEE Int. Conf. Humanoid Robots (Humanoids)*, Paris, France, Dec. 7–10, 2009, pp. 142–147.

[20] S. M. LaValle and J. Kuffner, "Rapidly-exploring random trees: Progress and prospects," in *Algorithmic and Computational Robotics: New Directions*, B. R. Donald, Ed. Wellesley, MA: A K Peters, 2001, pp. 293–308.

[21] Z. Xia, G. Chen, J. Xiong, Q. Zhao, and K. Chen, "A random sampling based approach to goal-directed footstep planning for humanoid robots," presented at the IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM), Singapore, Jul. 2009.

[22] Z. Xia, K. Chen, J. Xiong, and G. Chen, "Parameter optimization in non-uniform randomized footstep planning for biped navigation," in *Proc. 2009 IEEE Int. Conf. Mechatronics Autom. (ICMA)*, Changchun, China, Aug. 9–12, 2009, pp. 3745–3750.

Zeyang Xia (M'08) received the B.E. degree in mechanical engineering from Shanghai Jiao Tong University, Shanghai, China, and the Ph.D. degree in mechanical engineering from Tsinghua University, Beijing, China, in 2002 and 2008, respectively.

From 2005 to 2006, he was a Visiting Scholar at the Institute of Automatic Control Engineering, Technical University of Munich, Munich, Germany. From 2008 to 2009, he was a Research Fellow at the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore. He is currently a Postdoctoral Researcher in the Department of Mechanical Engineering, Indiana University-Purdue University, Indianapolis. His research interests include biped humanoid robot design, planning and control, medical robot design and motion planning, and orthodontic biomechanics and appliance development.

Jing Xiong received the B.S. and Ph.D. degrees in mechanical engineering from Tsinghua University, Beijing, China, in 2004 and 2010, respectively.

She is currently a Postdoctoral Researcher in the Department of Biomedical Engineering, Indiana University-Purdue University Indianapolis. Her research interests include design and surgical planning of interventional robots for minimally invasive therapy, motion planning for humanoid robots, and design theory of spatial mechanisms.

Ken Chen (M'93) received the B.S. degree in mechanical engineering from Sichuan University, Chengdu, China, in 1982, and the M.S. and Ph.D. degrees in mechanical engineering from Zhejiang University, Hangzhou, China, in 1984 and 1987, respectively.

From 1991 to 1992, he was a Visiting Professor at the University of Illinois, Chicago. From 1992 to 1995, he was a Postdoctoral Researcher at Indiana University-Purdue University, Indianapolis. He is currently a Professor and the Director of the Institute of Manufacturing Engineering, Department of Precision Instruments and Mechanology, Tsinghua University, Beijing, China. He has authored or coauthored more than 100 refereed papers published in several domestic and international academic journals and international conference proceedings. His research interests include robotics and intelligent control, humanoid robots, microrobots and small robots, medical and space robots, manufacturing automation systems, and hydraulic servo systems.

Dr. Chen is a member of the American Society of Mechanical Engineers and a Senior Member of the Chinese Mechanical Engineering Society. He was the recipient of the First Award of Chinese Youth Technology in 1988 and the Second Award of Technology Advancement from the People's Liberation Army.