2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Search-based Foot Placement for Quadrupedal Traversal of Challenging Terrain

Barrett Mitchell, Andreas G. Hofmann, and Brian C. Williams, *Member, IEEE*

***Abstract*—A primary motivation for employing quadrupedal robots is that their morphology allows them to traverse difficult terrain. For example, a mountain goat, by carefully choosing its foot placements, is able to scale steep cliff sides. In contrast, wheeled robots have difficulty traveling over non-level terrain, and bipedal robots face stability challenges on rough terrain, even at low velocities. In order for quadrupeds to perform traversals over rough terrain in a stable manner, robust navigation strategies are needed that allow the robots to take full advantage of their physical capabilities.**

Foot placement and body pose planning is one of the most challenging problems associated with such navigation. We approach this problem as a combinatoric search over candidate foot placements and body poses. The search returns the sequence of kinematically feasible steps with the lowest cost as determined by their deviation from the terrain-independent nominal steps. Due to the large search domain in this problem and the speed required by real time robots, searching for the true optimal solution is computationally intractable. Therefore, we use a limited-horizon best-first search that quickly finds a near-optimal feasible solution. We show, through a series of tests, that this algorithm is sufficient for traversing challenging terrain, with obstacle heights approaching the leg length of the quadruped.

I. INTRODUCTION

Over the past decade, a variety of robots capable of walking on level, un-obstructed terrain have been built [1]–[4]. The main purpose of these robots has been to demonstrate the feasibility of robotic walking. Recently, much work has been performed analyzing the efficiency and stability of these devices, particularly, passive dynamic and minimally actuated walkers [5]–[8]. The motivation behind this focus is to maximize energy efficiency. However, the stepping pattern of these devices is periodic, with little or no ability to adjust step foot placement or timing.

The primary reason for using legged robots, in the first place, is that their morphology should allow them to traverse difficult terrain, including terrain that cannot be covered by a wheeled vehicle, as shown in Fig. 1. One approach to controlling a legged device is to use compliant legs, without explicit control over foot placement [9]. This approach has allowed devices to traverse terrain with obstacles of significant size, relative to the robot. However, due to the lack of careful foot placement, this approach is only safe for relatively small robots, and the robot's body is not guaranteed to remain upright. For larger robots, it is desirable to control foot placement so that the robot remains balanced and posture is upright.

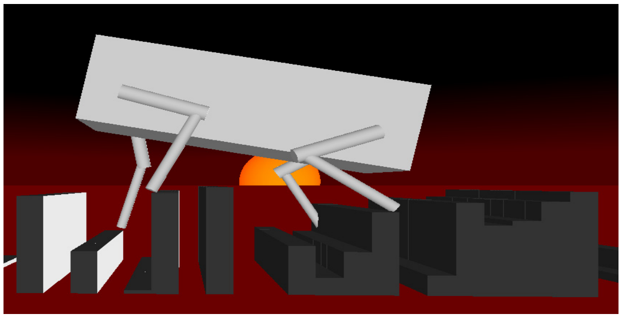

Fig. 1 – A quadruped can traverse difficult terrain that cannot be traversed by a wheeled vehicle of comparable size.

The problem of determining appropriate foot placements is challenging in that the quadruped is an articulated device with complex dynamics; foot placement choices must ensure that the quadruped is able to maintain its balance, that kinematic limits are not violated, and that the body and legs do not collide with obstacles in the environment.

Previous approaches to this problem [11, 17, 18], use a global search-based approach in which a potentially very large candidate set of footholds is searched to produce an optimal sequence from a start to a goal location. One reason that the candidate sets can be very large in these formulations is that the start and goal locations may be far apart, requiring many (dozens) of steps to get from start to goal. A second reason is that these formulations make no assumptions about leg stepping order; any leg is allowed to step at any time. Furthermore, these formulations are based on fundamental constraints, and make little use of heuristics, resulting in extensive, detailed searches of possible step footholds, and motion trajectories for each step.

These approaches are very impressive, from a theoretical standpoint, in that they produce optimal plans from first principles. However, the intensive computational requirements for these approaches make them ill suited for real-time control. We hypothesize that this is not the way humans and animals solve this problem, in most situations.

Manuscript received September 15, 2006.

Barrett Mitchell is with the Computer Science and Artificial Intelligence Laboratory (CSAIL) at MIT, Cambridge, MA 02139 USA; e-mail: bsmit@mit.edu.

Andreas G. Hofmann is with CSAIL and with the Media Lab at MIT; e-mail: hofma@csail.mit.edu.

Brian C. Williams is with CSAIL and with the Aeronautical and Astronautical Engineering Department at MIT; email: Williams@mit.edu.

1-4244-0602-1/07/$20.00 ©2007 IEEE.

We believe, rather, that a more local search incorporating key heuristics is adequate, and results in more modest computational requirements. In order to investigate this hypothesis, we have implemented an approach based on local search, with the following key features: 1) a direction vector indicating the general direction to the goal; 2) search over a limited, *receding horizon* of future steps; and 3) search by investigating progressively larger deviations from a nominal stepping sequence. Use of a limited horizon of future steps, and use of a heuristic that favors a particular nominal stepping sequence dramatically reduces the search space, and therefore, computational requirements. This is motivated by the hypothesis that humans and animals have a well-developed notion of nominal foot placement, and evaluate a relatively small number of possible foot placements when traversing difficult terrain; they use simple rules to quickly eliminate placements that will not work.

In order to determine whether this local search approach is adequate, we test our algorithm using a simulated quadruped, and an extensive, randomly generated set of obstacle courses. A key question addressed by these tests is how many future steps should be considered in the receding horizon.

II. METHODS

We begin by describing the test terrain and simulated quadruped. We then present details of the foot placement algorithm, and describe the evaluation tests performed.

A. *Test Terrain*

The test terrain consists of rectangular obstacles, as shown in Fig. 2. The obstacle height varies every 2cm in the forward direction but is constant in the lateral direction. For a particular test, a terrain instance is generated automatically, where the obstacle height at each 2 cm increment is selected randomly (with even distribution) to be between 0 cm, and a specified maximum obstacle height, which is a parameter of the test. The length of a terrain instance is always 72 cm, which is about three times the length of the quadruped's body.

Fig. 2 – The test terrain consists of rectangular obstacles, evenly spaced, but of random height.

B. *Simulated Quadruped*

A kinematic quadruped simulation was used to evaluate the foot placement algorithm. The quadruped's legs consist of two segments: an upper, and a lower leg segment. The upper leg segment is joined to the body with a two degree of freedom hip joint, allowing rotation about the forward and transverse axes. The lower leg segment is joined to the upper leg segment with a one degree of freedom rotational knee joint. Dimensions of the quadruped are shown in Table 1. The simple morphology of the legs allows the inverse kinematics to be computed analytically [12].

Body length	20.9 cm
Body width	11.3 cm
Upper leg length	7.3 cm
Lower leg length	8.7 cm

Table 1 – Quadruped dimensions

C. *Nominal Gait*

Nominal gaits for legged systems are derived based on optimal energy and stability considerations [15]. Such optimizations are computationally intensive, and are beyond the scope of this work. Therefore, we rely on the nominal gait to represent these desirable characteristics, and then seek to minimize deviation from this nominal gait.

The nominal quadrupedal gait, which is used on level terrain, in the absence of obstacles, is shown in Fig. 3. The feet stay 15.3 cm apart, and move forward in an alternating stepping motion, where the step size is 6 cm. Knees are slightly bent so that the body remains at a nominal height of 12 cm. The orientation of the body is upright.

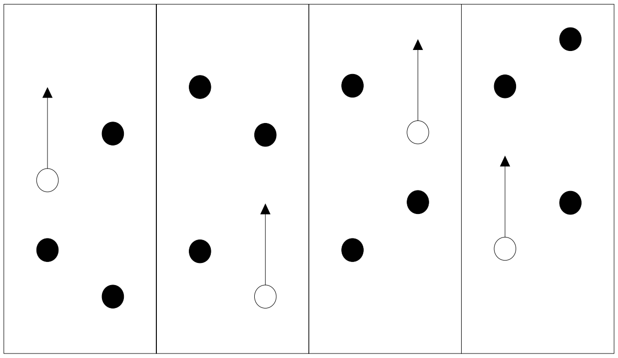

Fig. 3 – Nominal gait pattern: stepping with left front foot (a.), stepping with right back foot (b.), stepping with right front foot (c.), stepping with left back foot (d.).

At each step, the body is shifted laterally so that the ground projection of the quadruped's center of mass (CM) is always within the convex support polygon defined by the points where the feet are in contact with the ground. This helps to ensure that the quadruped remains in balance at all points in the nominal gait cycle.

D. *Foot Placement Algorithm*

Our approach to foot placement is based on the idea that the nominal foot placement is the best, and any deviations from this nominal placement should be minimized. Hence, we search through acceptable deviations, in order of increasing magnitude. We make the following assumptions: 1) the gait pattern is fixed; it is the pattern (sequence of steps) shown in Fig. 3; 2) the body orientation about the forward axis is zero, as in the nominal case; 3) the lateral and forward CM movement must be such that the ground projection of the CM remains within the convex polygon defined by the points where the feet make contact with the ground. The last assumption helps to ensure that the quadruped is able to maintain its balance when executing the steps [13]. Despite these restrictions, there is still significant opportunity for deviation from the nominal gait pattern. First, we allow the foot placement position to deviate from that in the nominal gait pattern. Thus, the stepping foot position may be further forward or behind, and may be to the side of the nominal position. Second, we allow the CM height to deviate from the nominal. Finally, we allow the body orientation about the transverse axis to deviate from 0. Thus, for each step, we search over a combination of foot placement positions for the stepping foot, CM heights, and body orientations. The problem can then be stated as one of finding an optimal sequence of foot placements, CM height, and body orientation combinations, given a terrain map, a current pose, and a receding horizon of n steps.

This can be expressed, more formally, as

$$\arg\min \sum_{i=1}^{n} \left((FP_{nom} - FP(i))^2 + (BH_{nom} - BH(i))^2 + (BR_{nom} - BR(i))^2 \right)$$

where $FP(i)$, $BH(i)$, $BR(i)$, are the foot placement, body height, and body rotation at step i, and FP_{nom}, BH_{nom}, BR_{nom}, are the corresponding nominal values. The values chosen for $FP(i)$, $BH(i)$, $BR(i)$, are subject to kinematic constraints. The kinematics of the quadruped can be expressed as

$$FP = A_{ll} A_k A_{ul} A_{hp} A_{hr} \, BP \qquad (1)$$

where $A_{ll}, A_k, A_{ul}, A_{hp}, A_{hr}$ are homogeneous transform matrices representing, respectively, translation of the lower leg, rotation of the knee, translation of the upper leg, and rotation of the hip. These transforms are nonlinear (trigonometric) functions of the leg joint angles. Thus, Eq. (1) is a nonlinear function of the joint angles that relates body pose, BP, and foot position, FP. A body pose and foot placement are kinematically feasible if there exists a set of joint angles such that Eq. (1) is satisfied.

The set of candidate foot placements is determined by a 1 x 20 cm grid with 1 cm spacing that is centered around the nominal foot placement, as shown in Fig. 4. Hence, the set allows for deviation from the nominal placement in the forward direction, but not in the lateral. This makes sense because for the test terrain, elevation does not vary in the lateral direction. Search of these candidate positions is performed in order of increasing deviation from the nominal position. We additionally restrict the set of candidate foot placements to ones that move the foot forward from its current position. Similarly, candidate body heights are chosen in the range of 3 cm above and below the nominal height, in increments of 1.5 cm, and candidate body rotations are chosen in the range of 15 degrees above and below 0, in increments of 7.5 degrees. Search over body heights and rotations is also performed in order of increasing deviation from the nominal.

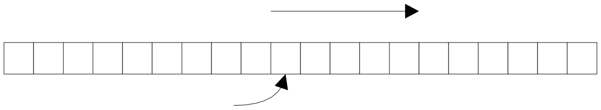

Fig. 4 – The foot placement grid defines candidate positions in the forward direction, centered on the nominal position.

Our algorithm uses best-first search, as shown in Fig. 5. The search queue, Q, is initialized to the current state and, at each iteration, the best node in Q is expanded. This expansion is accomplished by expandQHead, which first gets nominal foot placements for a given current state, and then iterates over combinations of candidate foot placements, body heights, and body orientations to generate a candidate child state. Each such combination is evaluated, using getCost, and the overall cost of the search path is added to the child state. When the expansion reaches sufficient depth, as indicated by numLookAheadSteps, the search terminates, returning the sequence of states leading to the expanded child state.

For each combination of foot placement, body height, and body rotation, a corresponding horizontal body position is computed that satisfies the constraint that the ground projection of the body CM remains within the support base. Using these values for body pose and foot position, a candidate state is constructed. This state is then evaluated using a feasibility checker, implemented by checkFeasibility in Fig. 5. The feasibility checks ensure that the state is kinematically feasible and does not result in contact of the legs with an obstacle. The first feasibility check uses the analytic inverse kinematic function to check that the candidate foot placement and body pose can actually be achieved using an appropriate set of joint angles (that Eq. 1 is satisfied). The second feasibility check determines whether the pose would result in contact of any of the legs with a terrain obstacle. This check is performed using geometry similar to that used to compute the inverse kinematics. It checks whether the upper and lower leg segments intersect the boundaries of obstacles in the vicinity of the foot placement. If both feasibility checks are satisfied, and if the required search depth, in terms of look ahead steps has been achieved, then the search terminates successfully.

```
nextState = FindNextState(baseState,n)
Q.insert(baseState)
Q.foundNextState = false
while Q.isNotEmpty
    Q = expandQHead(Q, Q.first)
    if(Q.foundNextState == true)
      nextState = Q.first
      return
    end
end

Q = expandQHead(Q, currState)
  Q.remove(currState)
  nFP = getNominalFootPlacement(currState)
  foreach cFP ∈ CandidateFootPlacements
    foreach bodyRotation ∈ CandidateBodyRotations
      foreach bodyHeight ∈ CandidateBodyHeights
           state = constructCandidateState(currState, cFP,
                              bodyRotation, bodyHeight);
           state.feasible = checkFeasibility(state, nFP,MAP);
           if isFeasible(state)
             Q = Q.insert(Q, state)
             if(state.lookAheadStep ==
                numLookAheadSteps)
              Q.foundNextState = true
              return
             end
           end

       end // bodyHeight
     end //bodyRotation
   end //cFP
```

Fig. 4 – Best-first search pseudo-code.

Fig. 6 provides an example of how increasing the number of look ahead steps improves performance.

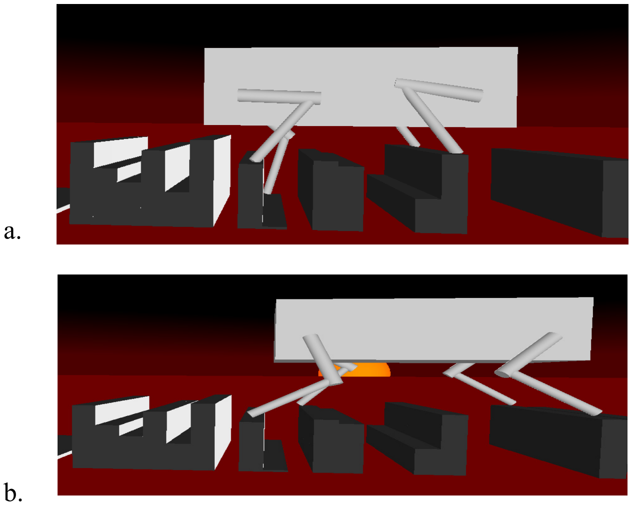

Fig. 6 – a. One step look ahead results in failure because the left rear leg has stepped into a deep hole; b. two step look ahead succeeds by avoiding the hole.

In Fig. 6a, one step look ahead is used, and the quadruped has landed in a state where the left rear leg is in a deep hole, between two obstacles. Furthermore, there is a large gap that the front two legs must traverse. Forward progress cannot be made from this position because in order for the front legs to traverse the gap, the body must move forward. However, moving the body forward would result in the left rear leg colliding with the obstacle in front of it. There are no feasible next steps from this position, and the quadruped is stuck. In Fig. 6b, two step look ahead is used. This allows the quadruped to avoid the deep hole; by looking two steps ahead, the quadruped knows that this leads to infeasibility. Instead, it places the leg on the tops of the highest obstacles, and is able to traverse the gap.

E. Test Description

To evaluate the foot placement algorithm, we performed a series of tests over a variety of terrain. First, the maximum obstacle height was varied from 3 cm to 10 cm, in increments of 1 cm. For each maximum obstacle height, 15 terrain maps were randomly generated. Each terrain map was 72 cm long, requiring about 50 steps to traverse.

For each terrain map, the number of look-ahead steps was varied from 1 to 6. The test for a particular number of look-ahead steps was then performed by initializing the quadruped in a nominal pose at the start of the course, and then successively generating foot placements, and executing them, until the quadruped had traversed the course, or until it failed due to infeasibility. Infeasibility occurs if Q becomes empty before a feasible search path has been found. The receding horizon control algorithm for traversing the course is shown in Fig. 7.

```
success? = TraverseCourse(n_look_ahead)
 current_state = initializeRobot;
 while (course_not_completed)
   nextState = FindNextState(current_state, n_look_ahead);
   if (nextState not feasible) return false;
   TakeFirstStep(nextState);
 end
 return true;
```

Fig. 7 – Receding horizon control for course traversal.

A key aspect of the receding horizon approach is that at each step, FindNextState generates a sequence of n_look_ahead steps. If this sequence is feasible, the first of these steps is executed, and the process repeats.

III. RESULTS

Fig. 8 shows a typical motion sequence for the quadruped. As can be seen from this sequence, a variety of foot placements, body heights, and body rotations are used to traverse the terrain.

Test results are summarized in Fig. 9. This shows the percentage of times the quadruped was able to successfully traverse a course with a particular maximum obstacle height, and for a particular number of look ahead steps.

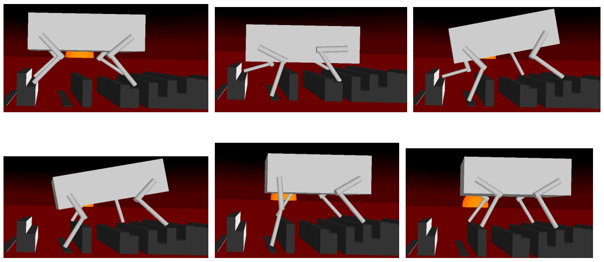

Fig. 8 – Typical motion sequence for terrain traversal.

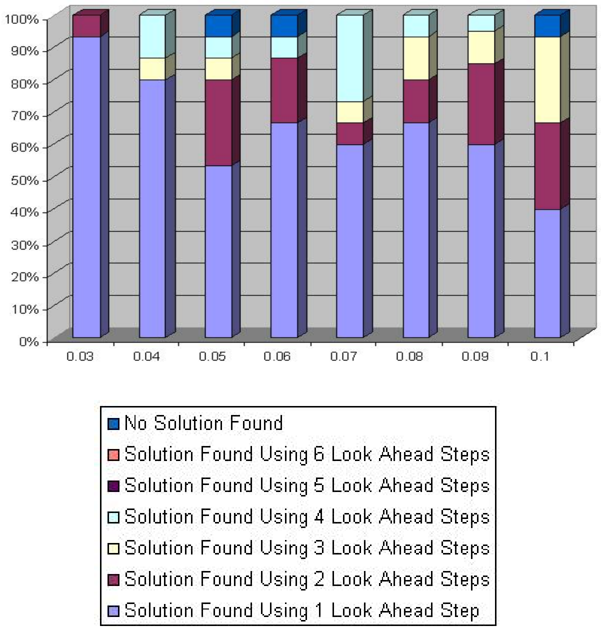

Fig. 9 – Success percentage for N look ahead steps, for obstacle heights ranging from 3 to 10 cm.

For example, for terrain maps with a maximum obstacle height of 3 cm, a successful traversal was achieved 90 % of the time using just 1 look ahead step, and 100% success was achieved using 2 look ahead steps. For terrain maps with a maximum obstacle height of 5 cm, a significant percentage of cases required 3 or 4 look ahead steps, and a small percentage could not be solved, even using 5 or 6 look ahead steps.

Note that the random terrain generation process does not guarantee that a particular instance can be traversed. Thus, it is possible that some of the cases that could not be solved with 6 look ahead steps, could not be solved at all, even with a large number of look ahead steps.

IV. DISCUSSION

A. Discussion of Results

The data shown in Fig. 9 suggests that a very limited search, beginning with a nominal solution, and progressing to ever increasing deviations from that nominal solution, can overcome difficult terrain, even if obstacle heights are significant. In these tests, a 10 cm obstacle height is significant because it is more than one half the length of the robot's leg.

These results support a number of conclusions. First, a significant percentage of terrain maps can be traversed using a small number of look ahead steps. In particular, half of the maps, including ones with 10 cm maximum obstacle heights, were successfully traversed using only 1 look ahead step. Second, the number of look ahead steps needed tends to increase as difficulty, indicated by maximum obstacle height, increases. Third, the marginal improvement gained from using more than 4 look ahead steps seems to be small.

The second conclusion is to be expected, but the first and third are noteworthy in that they suggest characteristics of this problem that must be considered when designing solution algorithms. In particular, for terrain were moderately sized obstacles are distributed in an even manner, as was the case for the tests we performed, adequate solutions are found without searching many steps into the future. In other words, a solution found using a look ahead step number of 20 is not much better than a solution found using a look ahead step number of 4. This is in contrast to problems like those encountered in chess, where optimal solutions may be sparsely and unevenly distributed, and where searching 20 moves ahead instead of 19 can make a critical difference.

What if there are large obstacles in the terrain that can't be traversed? A human hiker, when confronted with such an obstacle (a steep mountain, for example), will typically plan a route around it. However, this planning does not involve detailed step planning, only an overall assessment of sequences of waypoints (go west for ½ mile, north for 1 mile, etc.). This suggests that a decoupling of global and local navigation problems is desirable. Global navigation involves planning paths around major obstacles that are thought to be non-traversable, without detailed planning of foot steps. Local navigation then involves planning foot steps, over a very limited horizon, that seek to follow the path suggested by the global navigation. Such a decoupling avoids detailed planning of foot steps over long horizons.

B. Future Work

A notable anomaly in Fig. 9 is the fact that for the 4 cm maximum obstacle height case, solutions are found using 1, 3, and 4 look ahead steps, but none are found using 2. This suggests that the 15 randomly generated terrain instances for each maximum obstacle height may not be enough to draw rigorous, statistical conclusions. More testing, with more terrain instances, will be needed to further confirm and validate the results shown in Fig. 9.

Although the simulated quadruped used in these tests has a full, three-dimensional kinematic model, the tests described here are, in certain ways, one-dimensional. As shown in Fig. 4, the candidate foot placements include positions in front of and behind the nominal one, but not to the side. Future testing will use a more rectangular mesh than the one in Fig. 4, to allow for sideways deviations from the nominal position. To exercise this capability, obstacle height will be made to vary in the lateral direction, as well as in the forward direction.

The rectangular obstacles used in our tests are challenging in that the terrain instances generated cannot be traversed by a wheeled vehicle of size comparable to the quadruped. However, natural terrain includes not only rectangular obstacles, but also sloped terrain. Foot placement for such terrain must include consideration of the slope and contact friction of the foot with the ground, in order to avoid foot placements that result in the foot slipping.

The tests performed here check kinematic constraints, but they do not consider timing or dynamics of the robot. Kinematic checks are sufficient if the robot moves slowly; the requirement that the ground projection of the CM remain within the base of support provides a static stability guarantee that ensures that the robot will not fall down, if it moves slowly enough. As the quadruped moves more quickly, dynamic considerations need to be taken into account. In particular, motion trajectories for all joints that satisfy dynamic constraints as the mechanism moves between foot placements must be found. Rather than performing detailed runtime searches, as in [11, 17, and 18], our approach to this problem is to precompile sets of feasible trajectories for stepping motions into *flow tubes* [10, 14], and then to assemble these flow tubes into maneuver sequences corresponding to step sequences, as part of the local search.

Although extensions to more terrain types and to more dynamic movement complicate the search, we expect that the adequacy of a limited local foot placement search will be preserved as the extensions are implemented. Thus, a key feature of our future work is to continue decoupling global and local navigation, in order to preserve the key advantage that considerably fewer nodes are searched. Fewer nodes implies a lower computational requirement, and thus, better real-time performance. While this decoupling may lead to sub-optimal, and even infeasible solutions, we believe the level of sub-optimality will be small, and that infeasibility will be relatively rare. Furthermore, due to sensor limitations in many practical applications, a detailed terrain map from start to goal will not be available; detailed foot placement will necessarily be based on a local map around the vicinity of the quadruped. Finally, the prospect of occasional infeasible solutions in the local search does not imply mission failure. Just as a human hiker backtracks when reaching a dead end, so the quadruped could physically backtrack and plan again with more look ahead steps.

The tests presented here suggest that further work in the area of heuristics to guide and limit the search for foot placements would be of great value. For example, a heuristic that avoids stepping into deep holes would allow for solution of problems such as the one depicted in Fig. 6 with fewer look ahead steps. Such heuristics would likely be terrain dependent, so adaptive and learning techniques that quickly identify a small number of candidate foot placements should be investigated.

The tests performed here, even with their simplifying assumptions, reveal interesting aspects regarding the nature of the foot placement problem, which must be considered in evaluating future research directions.

REFERENCES

[1] K. Hirai, “Current and Future Perspective of Honda Humanoid Robot,” *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Grenoble, France, 1997, pp. 500 – 508.

[2] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, “The development of the Honda humanoid robot”, *IEEE International Conference on Robotics and Automation*, 1998.

[3] J. Yamaguchi, E. Soga, S Inoue, and A. Takanishi, “Development of a bipedal humanoid robot – control method of whole body cooperative dynamic biped walking”, *IEEE International Conference on Robotics and Automation*, 1999.

[4] S. Kagami, F. Kanehiro, Y. Tamiya, M. Inaba, and H. Inoue, “Autobalancer: an online dynamic balance compensation scheme for humanoid robots”, *Robotics: The Algorithmic Perspective*, B. R. Donald, K. M. Lynch, D. Rus, editors, A. K. Peters Ltd.., 2001, pp. 329 – 340.

[5] A. Goswami, B. Espiau, and A. Keramane, “Limit cycles and their stability in a passive bipedal gait”, *IEEE International Conference on Robotics and Automation*, 1996.

[6] S. Collins, M. Wisse, and A. Ruina, “A three-dimensional passive-dynamic walking robot with two legs and knees”, *International Journal of Robotics Research*, 2001

[7] S. Collins, A. Ruina, R. Tedrake, and M. Wisse, “Efficient bipedal robots based on pssive-dynamic walkers,” *Science*, 307, February, 2005, pp. 1082–1085.

[8] E. R. Westerveldt, G. Buche, and J. W. Grizzle, “Inducing dynamically stable walking in an underactuated prototype planar biped,” *IEEE International Conference on Robotics and Automation*, 2004

[9] R. Altendorfer, E. Z. Moore, H. Komsuoglu, M. Bueler, H. B. Brown Jr., D. McMordie, U. Saranli, R. Full, and D. E. Koditschek, “RHex: A Biologically Inspired Hexapod Runner,”Autonomous Robots 11:207-213, 2001

[10] A. G. Hofmann, B. C. Williams, “Exploiting Spatial and Temporal Flexibility for Plan Execution of Hybrid, Under-actuated Systems”, AAAI, 2006

[11] J. Chestnutt, M. Lau, J.J. Kuffner, G. Cheung, J. Hodgins, and T. Kanade. “Footstep Planning for the ASIMO Humanoid Robot”, in Proc. IEEE Int’l Conf. on Robotics and Automation (ICRA), 2005

[12] J. Craig. “Robotics”, Addison-Wesley, 1989

[13] M. Popovic, A. Goswami, and H. Herr, “Ground Reference Points in Legged Locomotion: Definitions, Biological Trajectories, and Control Implications,” *International Journal of Robotics Research*, 2005

[14] A. G. Hofmann. “Robust Execution of Temporally Flexible Plans for Bipedal Walking Devices”, Ph.D. Thesis, MIT, 2005

[15] Andersen and Pandy

[16] R. P. Paul. “Robot Manipulators”, MIT Press, 1981

[17] K. Hauser, T. Bretl, J. Latombe, and B. Wilcox. “Motion planning for a six-legged lunar robot”, *Workshop on Algorithmic Foundations of Robotics* (*WAFR*), New-York, NY, July 2006.

[18] T. Bretl. “Motion Planning of Multi-Limbed Robots Subject to Equilibrium Constraints: The Free-Climbing Robot Problem”, I*nt. J. of Robotics Research*, 25(4):317-342, Apr 2006.