

Planning energy-efficient bipedal locomotion on patterned terrain

Ali Zamani¹, Pranav A. Bhounsule¹, Ahmad Taha²
corresponding author: pranav.bhounsule@utsa.edu

¹ Dept. of Mechanical Engineering,

² Dept. of Electrical and Computer Engineering,
The University of Texas San Antonio,
One UTSA Circle, San Antonio, TX 78249, USA.

In Proc. SPIE 9837, Unmanned Systems Technology XVIII, Baltimore, Maryland, USA, April 20-21, 2016.

Abstract

Energy-efficient bipedal walking is essential in realizing practical bipedal systems. However, current energy-efficient bipedal robots (e.g., passive-dynamics-inspired robots) are limited to walking at a single speed and step length. The objective of this work is to address this gap by developing a method of synthesizing energy-efficient bipedal locomotion on patterned terrain consisting of stepping stones using energy-efficient primitives.

A model of Cornell Ranger (a passive-dynamics inspired robot) is utilized to illustrate our technique. First, an energy-optimal trajectory control problem for a single step is formulated and solved. The solution minimizes the Total Cost Of Transport (TCOT is defined as the energy used per unit weight per unit distance travelled) subject to various constraints such as actuator limits, foot scuffing, joint kinematic limits, ground reaction forces. The outcome of the optimization scheme is a table of TCOT values as a function of step length and step velocity. Next, we parameterize the terrain to identify the location of the stepping stones. Finally, the TCOT table is used in conjunction with the parameterized terrain to plan an energy-efficient stepping strategy.

1 Introduction

Bipedal robots inspired from passive-dynamics are highly energy efficient because they use their natural dynamics for walking [5]. So far these robots have been limited to walking on flat ground at a single speed and step length. For these robots to be able to walk in human-like environments, they would have to be able to traverse patterned terrain consisting of ditches, stairs, and obstacles. The challenge then is to create motion plans for these robots to enable them to walk on patterned terrain without sacrificing their high energy-efficiency. In this paper, we present some preliminary results towards this objective.

Motion planning of humanoid robots on patterned terrain has been studied before. The core idea is to first create stable motion primitives and then to use these primitives with a motion planner (e.g., Dynamic programming [9], A-star search [4]) to create motion plans.

To create energy-efficient motion plans, Kim et al. [8] created energy-efficient motion primitives parameterized by initial state and footstep sequence. These primitives were then used with an A-star search algorithm for motion planning. Huang et al. [7] used human walking data to create parameterized stepping costs. This step cost was parameterized by step length, step velocity, step width, and turning angle. A terrain cost parameterized by slope, altitude, roughness and bumpiness was added to the step cost. The sum of the step cost and terrain cost were used with an A-star search to create energy-optimal motion plans. Brandao et al. [3] followed a similar approach as Huang et al., but they used a physics simulation during the motion planning to ensure that the solutions were feasible in the presence of slippery surfaces.



Figure 1: Conceptual framework for bipedal locomotion on patterned terrain

The only study that used a passive dynamics-inspired model for motion planning is by Manchester and Umenberger [10]. They used a 2D model to create a library of non-steady controllers using Hybrid Zero Dynamics (HZD). This library was catalogued by the state of the system before impact, step length, and step height. Then the library was used in conjunction with a best-first search algorithm to find a feasible motion plan. No energy minimization was done in this study.

Our methodology is similar to the above study except that we explicitly minimize the energy during the motion planning. Fig. 1 shows our methodology. We generate a table of energy-efficient walking controllers parameterized by gait features such as step-length and step-velocity. This computation is done offline. We use the table of controllers, an energy-based optimization criteria, and terrain information to create a motion plan. Finally, we use the motion plan to generate a motion controller for the robot. Both these blocks are implemented online. In this paper, we will only focus on generation of table of controllers and motion planning (grey blocks). We present the details next.

2 Methods

2.1 Robot Model

We provide a brief review of the robot model. For more details refer to Bhounsule [1]. Fig. 2 (a) and (b) shows the robot and a 2D model of the robot. The 2D robot model has two legs. Each leg is characterized by a link length, mass, location of the center of mass (COM) and polar moment of inertia about COM. The feet are mass-less, circular and have point contact with the flat, rigid ground. The hips are driven by a single DC motor. There is a hip spring in parallel with the hip motor that has a linear spring stiffness and a set point corresponding to zero hip angle. There is an ankle motor on each leg that is connected to the ankle via a linear stiffness spring. The DC motor has coulomb and viscous friction losses. The parameters for the robot and the DC motor are identified in bench experiments.

A single step for the robot consists of a single stance phase (one leg in contact with the ground), followed by foot-strike when the swinging leg makes contact with the ground, followed by double stance (both legs on the ground), and back to the single stance phase. The equations describing single and double stance phase have continuous dynamics and are obtained using angular momentum balance about appropriate joints. The equations of motion for the foot-strike are algebraic equations and are obtained by conservation of angular momentum before and after foot-strike.

2.2 Energy-optimal trajectory control

The energy-optimal trajectory control problem is to find the most energy-efficient steady state walking solution for a given combination of step length and step velocity.

The energy-optimal control problem formulations is as follows. Given the walking sequence described earlier, find the hip and the ankle motor currents as a function of time that would minimize the energy metric called the Total Cost Of Transport (TCOT),

$$\text{TCOT} = \frac{\text{total battery energy used per step}}{\text{weight} \times \text{step length}}. \quad (1)$$

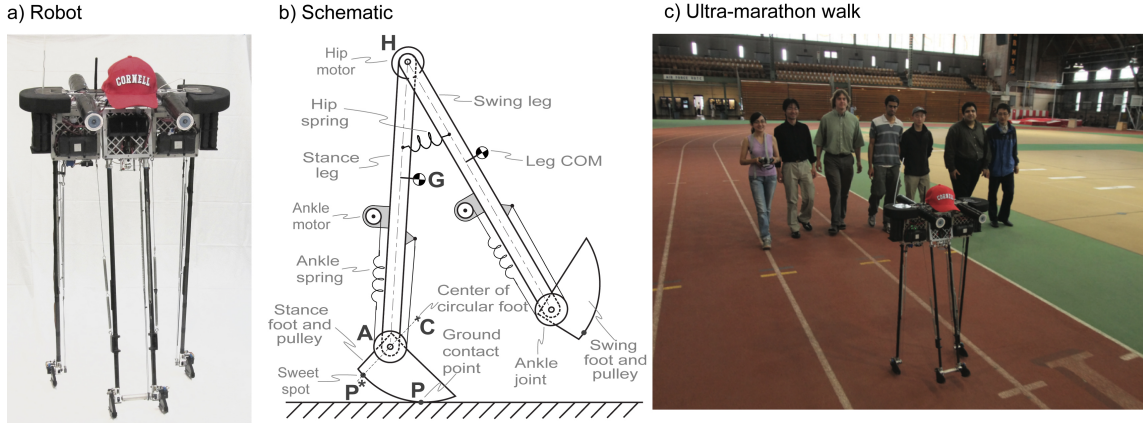


Figure 2: (a) Cornell Ranger. (b) 2D model used for energy-optimal control (c) Ranger during its ultra-marathon walk. On May 2, 2011, Ranger walked 40.5 miles non-stop on a single battery charge [2]. The walking was limited to flat grounds at approximately a single speed and step length.

The numerical trajectory optimization is constrained to periodic gaits, so we only optimize over one step and impose the periodicity constraint that the state (angles and rates) at the beginning of single stance should be equal to the state at the end of double stance. The other constraints are: the swinging leg's foot in single stance should have sufficient ground clearance to prevent foot scuffing; the motor currents must be within specified bounds to prevent overheating and damage to gear boxes; and tensional contact with the ground is forbidden. We use the sequential quadratic programming package SNOPT [6] to do solve the trajectory control problem.

For the motion planning described in the next section, we are interested in generating a table of TCOT values as a function of step velocity and step length. To do this, we add step length and step velocity constraints to the above optimization. We then solve the energy optimization for various combinations of step length and step velocity as described in the results section.

2.3 Motion planning

The motion planning problem is to find a walking strategy that will allow the robot to cross a terrain consisting of stepping stones while minimizing the given cost function.

The motion planning problem formulation is as follows. Find a sequence of walking steps, $i = 1, 2, \dots, N$, each step is characterized by a step length (D_i) and a step velocity (V_i) that will allow the robot to cross a terrain with a sequence of stepping stones at known locations and of known width while minimizing the cost function given by

$$\text{Cost} = \text{TCOT}(v_1, d_1) + \alpha w (v_1 - v_0)^2, \quad (2)$$

$$\text{where } w = \frac{\text{TCOT}_{\max}}{(v_{\max} - v_{\min})^2} \text{ and } 0 \leq \alpha \leq 1,$$

where v_0 is the step velocity at the previous step, v_1 and d_1 , are the step velocity and step length respectively at the current step, v_{\min} , v_{\max} , TCOT_{\max} are the minimum and maximum step velocities and the maximum TCOT respectively as computed in the energy-optimal trajectory control problem, w is the normalization factor to appropriately scale the two costs, and α is a scalar such that, $0 \leq \alpha \leq 1$.

The first term in the cost function is the energy metric, TCOT. This has been evaluated in Sec. 2.2 but only for steady-state (periodic) walking motions. The second term is the square of the change in velocity between steps. This quantity is heuristically chosen and loosely correlates to the energetic cost of transitioning from one steady-state (periodic) walking motion to another. We vary α to understand how these two costs affect motion planning.

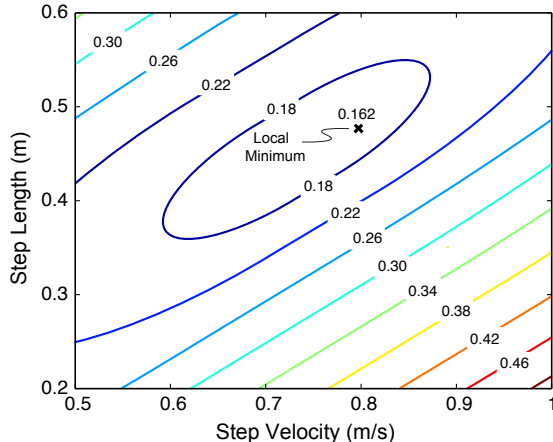


Figure 3: Contour plot of TCOT as a function of step velocity and step length.

We discuss the solution of the motion planning problem next. We generate a table for the cost function, Eqn. 2, as a function of v_0 , v_1 , and d_1 . We sort this table in increasing value of the cost function for a given α . Before taking a step, the robot scans the terrain up to distance equal to the maximum step length, d_{\max} . A grid of equal spacing is assigned to the terrain up to d_{\max} . If there is a ditch at a particular grid point, a value equal to zero is arbitrarily assigned, else a value equal to one is assigned. Using this value system, the robot is able to determine all possible places that it can take a single step. Next, the table is checked one row at a time starting from the first row to determine a feasible walking step. Since the table is sorted in the increasing value of cost function, the simple table lookup from top to down ensures that the walking strategy with the least cost is selected. The robot then takes a single step. This process of looking ahead, planning, and taking a step is repeated till the robot reaches the end of the terrain. In summary, the robot is following a one-step planning algorithm because it is planning a single step at a time. Our motion planning algorithm is a greedy search algorithm because it searches for a local optimum solution in the hope of finding a global optimum. For more details please see Alg. 1 in the Appendix.

3 Results and discussion

We present results for the energy-optimal trajectory control problem in Fig. 3. The local minimum has a TCOT = 0.162 at a step length of 0.48 m and a step velocity of 0.77 m/s. For this solution, 48% of the energy budget is used to power the computers and sensors on board and the remaining 52% is used by the ankle and hip motors to take a step.

We solve the energy-optimal solution for various combinations of step-length and velocity. The step length range is 0.2 to 0.6 m in increments of 0.1 m, and the step velocity range is 0.5 m/s to 1.0 m/s in increments of 0.1 m/s. Thus we performed a total of 30 optimizations (5 step length and 6 step velocity combinations). We use these for motion planning results shown next.

For all motion planning simulations here, we start the robot with an initial velocity, $V_0 = 0.5$ m/s. Fig. 4 (a) and (b) shows results for $\alpha = 0$ and for $\alpha = 0.8$. The top plot shows the terrain with stepping stones highlighted in gray. The planned footstep locations are shown with a blue star. The step velocity corresponding to a step is shown in the bottom panel. In (b), the robot walks with a step velocity of 0.5 m/s over the entire terrain to reduce the variation in speed to zero because the transition energy cost (second term in Eqn. 2) penalizes a change in speed. Finally, note that the strategy in (b) requires three more steps compared to (a).

Fig. 5 (a) shows a plot of TCOT as a function of α . As we increase α from 0 to 1 the TCOT increases

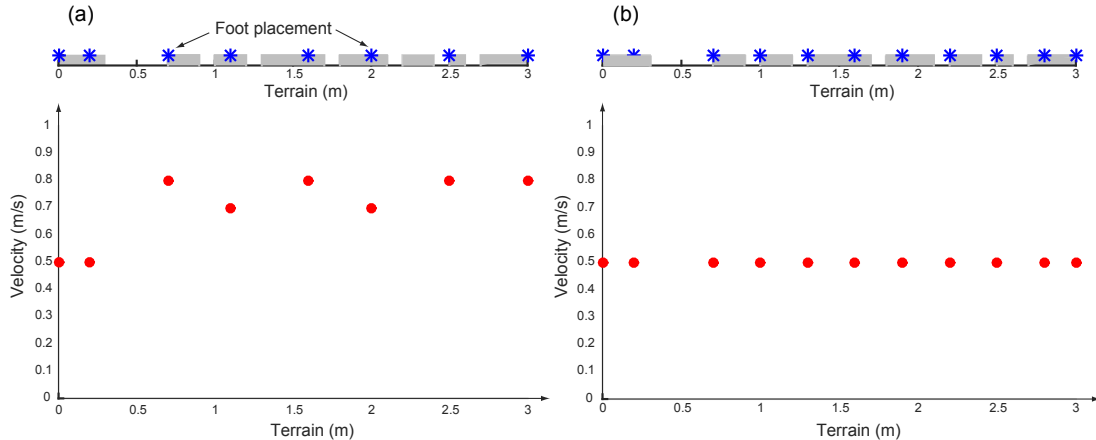


Figure 4: Strategy for walking on terrain with ditches for different weighting factors: (a) $\alpha = 0$ and (b) $\alpha = 0.8$. The plot on the top shows the terrain. The blue stars shows the foot stepping strategy. The plot on the bottom shows the step velocity.

from 0.1846 to 0.2130 (a 15% increase), as expected. As we increase α , the motion planning algorithm puts increasing emphasis on the cost of transitioning from one speed to another between steps. The net result is that a less optimal TCOT strategy is chosen. Fig. 5 (b) shows a plot of deviation in velocity as a function of α . As α increases, the motion planning algorithm chooses a more conservative strategy in which the change in speed between steps is minimized to zero and the number of steps is increased from 8 to 11 for $\alpha = 0$ and $\alpha = 1$ respectively.

How much more expensive is it to walk on patterned terrain? The most energy-efficient walking with no constraints on the step length or step velocity has a $\text{TCOT} = 0.162$. On patterned terrain, our motion planning algorithm was able to find a walking strategy with a $\text{TCOT} = 0.1846$. The increase in the TCOT is about 14%. This is because the robot motion is constrained to non-optimal step-lengths and step-velocities because of the stepping stones.

We have used a simple proxy for the energetic cost of transitioning during motion planning, which is the square of the difference in velocity between steps. But a more accurate solution would be to generate a table of non-steady walking solutions and use it for motion planning. But this could become computationally expensive for a high dimensional system. An alternate solution would be to create a Linear Quadratic Regulator (LQR) for transitioning between steady state (periodic) solutions [11]. The LQR cost can then be used for the transition cost.

The motion planning algorithm presented here is myopic as it only looks one-step ahead. The main issue with one-step look ahead is that it is less optimal than multiple-step look ahead. But the computational cost of multiple steps look ahead can be prohibitively high. It would be interesting to understand the tradeoff between storage/computational cost versus energetic cost as multiple steps are planned.

4 Conclusion

This study presented a motion planning algorithm for a passive-inspired robot, the Cornell Ranger. A trajectory control problem based on minimizing the energy to generate steady state walking was formulated and solved. The optimization produced a table of values of the energy used as a function of step length and step velocity. This table was used in conjunction with a greedy-search type motion planning algorithm to find a feasible motion while minimizing a cost function consisting of (1) energy needed for steady state locomotion found earlier and (2) heuristic for the energy needed to transition from one speed to another.

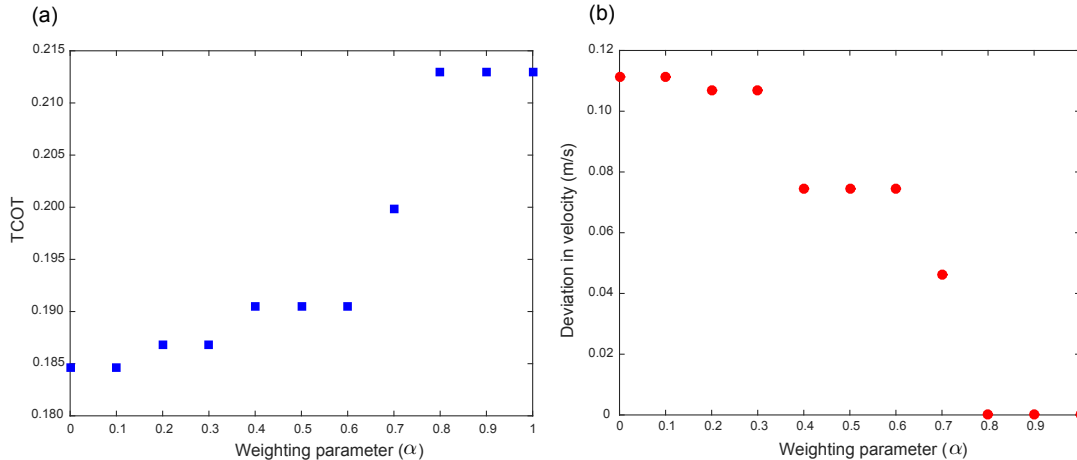


Figure 5: (a) TCOT and (b) standard deviation in the velocity. Both these are plotted as a function of the weighting parameter α .

The motion planning algorithm looks a step ahead for ditches and then uses the table of energy values parameterized by step-length and velocity to take a step. This process continues till the robot reaches the end of the terrain. The one-step planning algorithm was able to find a walking strategy on terrain with ditches while minimizing the energy-based cost function. When we increased the weight on the transition energy costs, the number of steps needed increased while the deviation in velocity between steps decreased to zero.

References

- [1] Pranav A. Bhounsule. *A controller design framework for bipedal robots: Trajectory optimization and event-based stabilization*. PhD thesis, Cornell University, 2012.
- [2] Pranav A. Bhounsule, Jason Cortell, Anoop Grewal, Bram Hendriksen, J.G. Daniël Karszen, Chandana Paul, and Andy Ruina. Low-bandwidth reflex-based control for lower power walking: 65 km on a single battery charge. *International Journal of Robotics Research*, 2014.
- [3] Martim Brandao, Kenji Hashimoto, Jose Santos-Victor, and Atsuo Takanishi. Optimizing energy consumption and preventing slips at the footstep planning level. In *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*, pages 1–7. IEEE, 2015.
- [4] Joel Chestnutt, Manfred Lau, German Cheung, James Kuffner, Jessica Hodgins, and Takeo Kanade. Footstep planning for the honda asimo humanoid. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 629–634. IEEE, 2005.
- [5] S. Collins, A. Ruina, R. Tedrake, and M. Wisse. Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307:1082–1085, 2005.
- [6] P.E. Gill, W. Murray, and M.A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Journal on Optimization*, 12(4):979–1006, 2002.
- [7] Weiwei Huang, Junggon Kim, and Christopher G Atkeson. Energy-based optimal step planning for humanoids. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 3124–3129. IEEE, 2013.

- [8] Junggon Kim, Nancy S Pollard, and Christopher G Atkeson. Quadratic encoding of optimized humanoid walking. In *Humanoid Robots (Humanoids), 2013 13th IEEE-RAS International Conference on*, pages 300–306. IEEE, 2013.
- [9] James J Kuffner Jr, Koichi Nishiwaki, Satoshi Kagami, Masayuki Inaba, and Hirochika Inoue. Footstep planning among obstacles for biped robots. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 1, pages 500–505. IEEE, 2001.
- [10] Ian R Manchester and Jack Umenberger. Real-time planning with primitives for dynamic walking over uneven terrain. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 4639–4646. IEEE, 2014.
- [11] R. Tedrake. LQR-Trees: Feedback motion planning on sparse randomized trees. In *Proceedings of Robotics Science and Systems (RSS), Zaragoza, Spain, 2009*.

APPENDIX

The motion planning algorithm is described below.

Algorithm 1 Motion planning algorithm

Input: energy metric (TCOT(v, d)), weighting parameter (α), initial velocity (V_0), terrain length (L).

Output: total steps (N), energy-metric (TCOT $_i$), step length (D_i), step velocity (V_i), where $i = 1, 2, \dots, N$.

- 1: Step counter ($i = 1$).
 - 2: Current foot location ($X = 0$).
 - 3: Generate a table, TABLE(rows,columns), with five columns:
 - 4: TABLE(:,1) = cost function (Cost),
 - 5: TABLE(:,2) = energy-metric (TCOT),
 - 6: TABLE(:,3) = step velocity at previous step (v_0),
 - 7: TABLE(:,4) = step velocity for current step (v_1),
 - 8: TABLE(:,5) = step length for current step (d_1).
 - 9: Sort TABLE in ascending order of first column, the Cost.
 - 10: Get maximum step length possible from TABLE, (d_{\max}).
 - 11: **while** $X \neq L$ **do**
 - 12: Identify all possible footholds, $x = \{x_1, x_2, x_3, \dots\}$, up to distance d_{\max} .
 - 13: index = 1.
 - 14: success = false;
 - 15: **while** success=false **do**
 - 16: **if** TABLE(index,3)= V_{i-1} and TABLE(index, 5) is in x **then**
 - 17: TCOT $_i$ = TABLE(index, 2),
 - 18: V_i = TABLE(index, 4),
 - 19: D_i = TABLE(index, 5)
 - 20: success = true;
 - 21: **else**
 - 22: index = index + 1;
 - 23: **if** index > length of TABLE **then**
 - 24: No solution exists, quit.
 - 25: **end if**
 - 26: **end if**
 - 27: **end while**
 - 28: $i = i + 1$
 - 29: **end while**
-