Efficient Robot Hopping Mechanisms with Compliance using Evolutionary Methods

Pramod Pal[1], Anubhab Dasgupta[2], Avinash Bhashkar[3], Shishir Kolathaya[1], and Ashitava Ghosal[1]

[1] Robert Bosch Centre for Cyber-Physical Systems, Indian Institute of Science, Bangalore, India `{pramodpal,shishirk,asitava}@iisc.ac.in`
[2] Dept. of Mechanical Engineering, Indian Institute of Technology, Kharagpur, India `anubhab.dasgupta@kgpian.iitkgp.ac.in`
[3] Dept. of Mechanical Engineering, Indian Institute of Technology, Kanpur, India `avinashb21@iitk.ac.in`

Abstract. This paper presents as a study of a single leg of a quadruped robot with a compliance mechanism in its foot. The objective is to optimize the hopping behavior in terms of maximum jump height, least impact force for a minimum jump height, and power consumed in the actuators of the leg as a function of the compliance in the foot. The paper consists of numerical simulations of the leg in the MuJoCo environment and experimental results obtained in hardware. In order to find the optimum, a two-stage algorithm is used. In the first stage, an exhaustive coarse grid search of the parameter space, consisting of the compliance in the foot and controller gains at the hip and knee actuators, is done to arrive at a set of parameters. Next, an evolutionary algorithm is used to conduct a more focused search beginning from the best set of parameter values found in the first phase. Simulation results show that there is an optimum value of the compliance in the foot, which can give efficient hopping behavior with the highest jump height and lesser impact force on hitting the ground. Experimental results obtained from hardware also show an optimum spring stiffness and are in reasonable agreement with the simulation results.

Keywords: Quadruped · Single legged robot · Evolutionary Optimization · Design

1 Introduction

Compared to wheeled mobile robots, walking robots are ideal for traversing uneven terrain due to their inherent versatility and ability to adapt to the environment. A legged robot can jump or go over obstacles, negotiate soft and hard terrains, flat or sloped terrains, and change directions or motion instantaneously. One of the key features of a walking robot is jumping or hopping, and in this paper, we study the hopping behavior of a leg which is a part of a quadruped. The robotic leg in this study consists of a waist, a hip joint, a knee joint, and a foot. In nature, animals have compliant elements in their limbs in the form of the tendon, muscles, etc., which are used to minimize impact forces and store energy while walking [1]. The compliance in a legged robot is similarly expected to absorb some of the impact force, store energy during the stance phase, and release energy during the swing the phase of locomotion, thereby increasing the efficiency, durability, and speed of the robot [2].

Raibert, a pioneer in the area of hopping robots, introduced a passive prismatic joint in the hopping robotic leg developed by him – the springy link of the hopping leg was of a telescopic type. Raibert and associates went on to develop mono-ped, biped, and quadruped and analyzed various gaits [3]. Ahmadi and Buehler worked on mono-ped with compliance and showed energy efficiency [4]. Hyon and Mita modeled a simple model of a hind limb to create a robot that could run like a dog with a tensile spring as a tendon [5]. In the work by Michele Focchi, the effect of different spring stiffness is studied by plotting the torque generated by the impact in the knee joint when the leg hits the ground [6].

To the best of our knowledge, limited literature deals with the design of compliant hopping mechanisms using learning methods [7]. The most recent work involves using Bayesian techniques for obtaining an optimal set of SLIP (spring-loaded inverted pendulum) parameters [8]. However, Bayesian methods are not scalable for large samples and parameters. In this work, we explore optimization methods in the context of compliant hoppers.

2 Design: Simulation and Hardware

The single leg is an essential component in a walking robot and in a quadruped for its mobility and stability. Designing a leg has the following essential components, link length, actuator placement, link shape, foot design, etc. In general, a shorter link length increases stability and control, as well as reduces weight, but restricts the range of motion and produces less force. On the other hand, a longer link length enhances the range of motion and power but diminishes stability and control and adds weight. In this work, we assumed uniform link lengths of 0.17 m, which is about the same as the leg length of a medium-sized dog. A well-designed leg/foot should be capable of absorbing impact force, which prevents its components from failing. Using spring at the foot can help in mitigating the impact force and, at the same time, store and deliver the energy while performing different gaits

2.1 Leg parameters

The single leg comprises a base, thigh, shank, and foot link connected with two active revolute joints and one passive prismatic joint (see Fig 1(a)). The thigh and shank leg lengths are denoted by l_1 and l_2, and the angles θ_1 (from vertical) and θ_2 (from the direction along the thigh link) denote the rotations of the links, and their ranges are shown in Table 1.

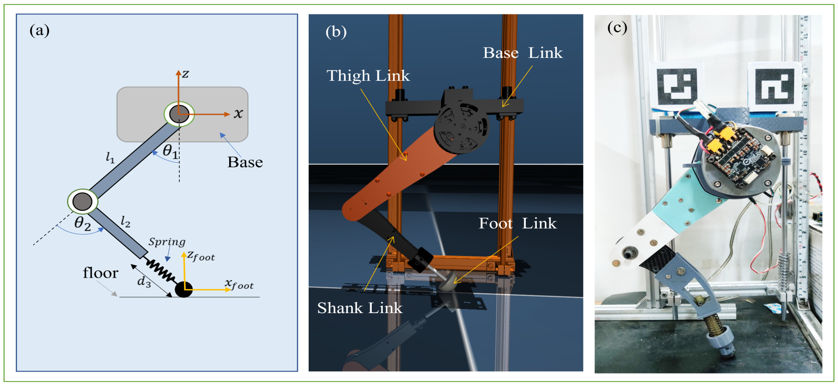


Fig. 1. (a)Schematic diagram of the single leg with compliant foot, (b) Single leg in MuJoCo environment, (c) Single leg hardware

Table 1. Motion range of the joint variables

Joint Variable	**Joint name**	**Type of joint**	**Range**
θ_1	hip	Revolute	0 to 1.6 radian
θ_2	knee	Revolute	-2.1 to 0 radian
d_3	foot	Prismatic	0 to 0.012 m

2.2 Simulation platform

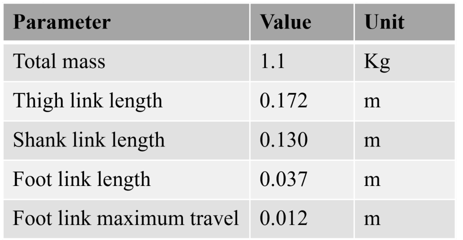
The simulations are performed using the Multi-Joint dynamics with Contact (MuJoCo) [9] simulation engine. MuJoCo captures contact dynamics with state-of-the-art capabilities. It is a popular open-source program for robotic system simulation and is simple to use. The model shown in the Fig 1(b) represents the single leg that has been designed using a CAD software (SolidWorks), and then the URDF (Unified Robotics Description Format) is created. The geometry and inertia parameters (from the CAD model) are then used for the simulation in MuJoCo. Simulation parameters, such as the step size, the number of iterations, choice of solvers are also made to govern the simulation. In the case of the leg, we have also chosen control parameters, such as actuator forces and torques, to define how the leg is to be controlled. MuJoCo also enables accurate visualization by choice of camera position and lighting conditions and environment parameters describe the attributes of the simulated environment, such as the coefficients of gravity and friction. All of these parameters work together to form a realistic depiction of the robot and its environment, enabling the precise testing and assessment of robotic systems.

We have used a few basic parameters to control the simulation of the leg – gravity allows the single leg to fall to make contact with the ground when it hits the ground plane, and we have used the proportional and derivative controller gain of a motor to simulate the behavior of a spring and damper as an added compliance mechanism to the foot of the single-legged robot. We have used a small time step of 0.001, which provides better accuracy and stability. For the solver, we have used Newton's method and a 4^{th}-order Runge-Kutta method. The total mass of the single leg is 1.1 kg. At all joints, we have used a damping of $1.0\,\mathrm{N\,s\,m^{-1}}$ and the torque is limited in both hip and knee joints to $7.0\,\mathrm{N\,m}$.

2.3 Experimental setup

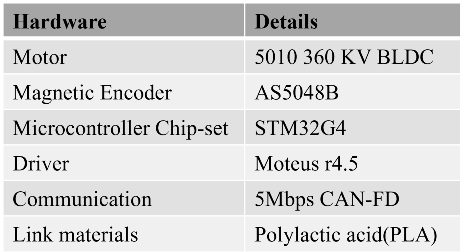
To validate the simulation results and to obtain the effect of the compliant foot, we have used hardware as shown in Fig. 1 (c). Both the hip and knee actuators were attached to the base link. The rotary motion of the knee motor is transferred to the shank link by a belt-and-pulley drive with a 1:2 gear ratio between the driver and the driven pulley. We've used a planetary gearbox with a gear ratio of 1:6 for power transfer from the motor to the links. The foot link is connected to the lower end of the shank link to the foot using a linear spring. Table 2 shows the parameters and their values, while Table 3 shows the hardware elements and their details. In the experiments, we have used different springs whose stiffness vary from 100 to $30\,000\,\mathrm{N\,m^{-1}}$. The power consumed by the leg was obtained from the voltage and the current supplied to the motor, and the height of the hopping was measured using an ultrasonic sensor with the least count of 0.05 cm.

Table 2. Single leg parameters.



Parameter	Value	Unit
Total mass	1.1	Kg
Thigh link length	0.172	m
Shank link length	0.130	m
Foot link length	0.037	m
Foot link maximum travel	0.012	m

Table 3. Hardware element and their details.



Hardware	Details
Motor	5010 360 KV BLDC
Magnetic Encoder	AS5048B
Microcontroller Chip-set	STM32G4
Driver	Moteus r4.5
Communication	5Mbps CAN-FD
Link materials	Polylactic acid(PLA)

3 Control Strategy, Optimization Objective and Training

In this section, we describe the details of the simulation and experiments done on the single leg with compliance at the foot.

3.1 Control Strategy

In this work, torque is the control input in both the hardware and the simulation. The torque is computed based on feedback from measured values, its first derivative, and its corresponding desired or reference values. We denote the hip, knee, and foot positions by the symbols x_h, x_k and x_f, respectively and their first derivatives by $\dot{x_h}, \dot{x_k}, \dot{x_f}$. The torque is computed as

$$\tau_h = k_{p_h} e_h + k_{dh} \dot{e_h}; \tau_k = k_{p_k} e_k + k_{dk} \dot{e_k}; \tau_f = k_{p_f} e_f \tag{1}$$

where, $e_h = \hat{x}_h - x_h$ is the error with $\hat{x}_h$ denoting the reference or desired value of the hip angle. A similar correspondence exists for e_k and e_f.

Inverse kinematics for reference trajectory generation: To make the leg hop in one place, a linear, cyclic trajectory is used to approximate the foot end-point trajectory in the $X - Z$ plane. This is given by

$$x = 0; z = -0.174 + 0.026 * \sin(\phi) \tag{2}$$

where the angle ϕ is used to divide the trajectory into 200 points. The three joint variables θ_1, θ_2 and d_3 for a given (x, z) in a leg are obtained using sequential least squares programming (SLSQP) algorithm [10]. The quantity $||\mathbf{q}||_2{}^2$ is used as the optimization objective function, where $\mathbf{q}$ denotes the vector of the hip angle, knee angle, and the change in length divided by the original length of the spring-damper system, to obtain the inverse kinematics solution of the compliant legged system. Once the reference trajectory is generated, respective torque values are calculated, and the trajectory of the single leg can be observed.

3.2 Data-driven evolutionary optimization for training parameters

The motion of the leg is a function of five controller gains – proportional and derivative gains at the hip and knee motors and the stiffness of the spring at the foot. The jump height and the impact force as the leg hits the ground is a function of these 5 parameters, and symbolically they can be expressed as:

$$h = f_1(\mathcal{K}); \mathcal{F} = f_2(\mathcal{K}) \tag{3}$$

where h denotes the hopping height, $\mathcal{F}$ denotes the highest impact force experienced by the foot over a cycle of the simulation and $\mathcal{K}$ denotes the vector of the controller gain values and the stiffness of the foot.

The dynamics of the foot impacting the ground is fairly complex and the task of finding the functional representation above is not tractable. In our work, we have used the MuJoCo enviroment to simulate the motion of the leg as a function of the 5 parameters. We use a two-step approach to obtain the optimum values of the 5 parameters. We first perform a "grid-search", where a sufficiently large range of the parameters is chosen, and an exhaustive search is performed where the program iterates through small step sizes in the space and records the value of the objective function at those nodal points. Following that comes the second step of our algorithm, where a more refined search is done using the Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES) [11] only around a selected nodal point to find the best objective, which minimizes the impact force or maximizes the height. Hence, we solve two optimization problems:

$$\max_{\mathcal{K}} \quad Ce^{-(H-h)}$$

and

$$\min_{\mathcal{K}} \quad \mathcal{F}$$

Both are subject to system dynamics and in the second, a penalty term is added if the height falls below a particular value. This is done to ensure that while still minimizing the impact force, the jump height does not become poor. In the above optimization problems H is an assumed constant upper bound on the jump height and C is a constant.

3.3 Training parameters

For the grid search, we used a range of about 500 to 1300 $\mathrm{N\,m\,rad^{-1}}$ for the proportional controller gains and small derivative gains ranging from 0.01 to 1.5 $\mathrm{N\,m\,s\,rad^{-1}}$ for the hip and knee motors. The spring stiffness for the foot is varied between 500 to 5000 $\mathrm{N\,m^{-1}}$. For the objective function in CMA-ES, we took 4000 steps and calculated the objective as the maximum of objective values over an entire cycle of simulation. The motivation for choosing the pointwise maximum is well known in optimization literature for it preserves the nature of the problem.

4 Results and discussion

In this section, we present the numerical simulation and the experimental results.

4.1 Simulation Results and Analysis

The single-leg hopping is simulated in MuJoCo. Fig. 2 (a) shows the convergence of the optimization process for a typical value of the spring stiffness. The maximum height with number of training steps converge to about 0.08 m. Figure 2 (b) shows the impact force with number of simulation steps – the impact force converges to a minimum of about 70 N. Figure 2(c) shows the jump height with increasing number of simulations – after the initial transient, the periodic behavior is independent of the initial starting position of the leg, which demonstrates the stability and robustness of the controller with respect to initial joint states. The average jump height is computed from this figure for different values of spring stiffness and Fig. 2 (d) shows the variation of jump height with spring stiffness. Figure 2(d) clearly shows that there exists an optimal spring stiffness to obtain maximum jump height jump for other parameters being fixed.

4.2 Experimental Results and Analysis

The experiments were done on a hard surface. Fig. 3 (a) shows the percentage of hopping height to the maximum range vs the spring stiffness. The result for the fixed foot was obtained when the spring was removed, and the foot was rigidly connected to the shank. As shown in Fig. 3 (a), the near-optimal hopping

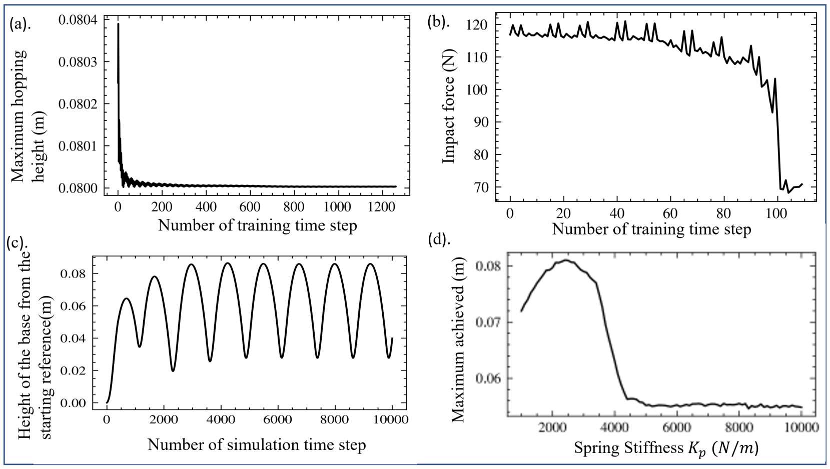


Fig. 2. Average hip and knee motor combined power versus foot stiffness

height range is obtained for spring with approximate stiffness $2800\,\mathrm{N\,m^{-1}}$. The hopping height decreases on both sides of this value - this variation of the hopping height range with spring (feet) stiffness and the existence of an optimum value of the spring stiffness is consistent with the numerical results shown in Fig. 2. It can also be seen from Fig. 3 (b) that the nature of the power consumed plot shows a minimum value for spring stiffness of $2800\,\mathrm{N\,m^{-1}}$. This shows with optimal stiffness, we achieve the near-optimal hopping height along with low power consumption. The experimental results are in reasonable agreement with the simulation results obtained using MuJoCo.

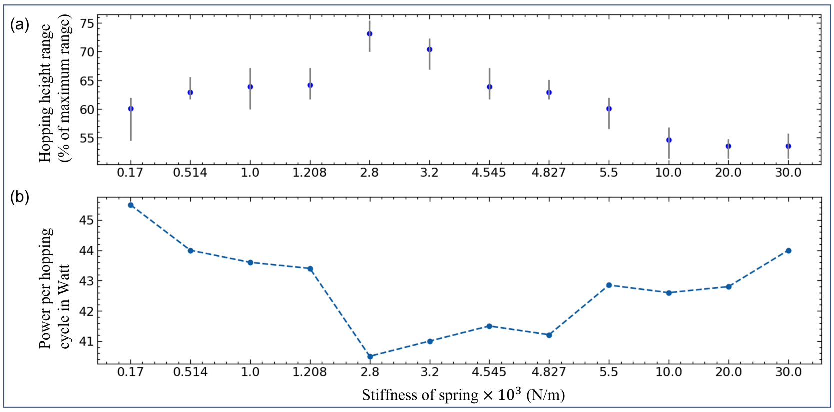


Fig. 3. (a) Plot of Hopping height range (%of Maximum range) to spring stiffness ; (b) Plot of Power consumed per hopping cycle to different spring stiffness

.

5 Conclusion

This paper presents a study of the effect of compliance in the foot of a single legged hopping robot. The single leg contains two motors and the goal is to obtain an optimal set of parameters for maximum vertical jump and minimum power consumption using evolutionary strategies. A model of the single legged robot is analysed using MuJoCo and a physical hardware with similar mass and geometrical parameters is fabricated. The main results of the work are a) there exists an optimum range of the stiffness and control gains which results in the highest vertical jump, b) the power required to jump is lower for this stiffness, and c) the experimental results obtained from the hardware are consistent with the simulation results. This work is being extended to training using the actual hardware using model free reinforcement learning techniques. We also plan to extend this work to a quadruped where stiffness will be added to each of the four legs and investigate the effect of the compliance on the feet towards cost of transport for the quadruped.

References

1. Hyon SH, Emura T, Mita T. Dynamics-based control of a one-legged hopping robot. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering. 2003 Mar 1;217(2):83-98.
2. Degrave J, Caluwaerts K, Dambre J, Wyffels F. Developing an embodied gait on a compliant quadrupedal robot. In2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2015 Sep 28 (pp. 4486-4491). IEEE.
3. Raibert M, Chepponis M, Brown HB. Running on four legs as though they were one. IEEE Journal on Robotics and Automation. 1986 Jun;2(2):70-82.
4. Ahmadi M, Buehler M. The ARL monopod II running robot: Control and energetics. InProceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C) 1999 May 10 (Vol. 3, pp. 1689-1694). IEEE.
5. Hyon SH, Mita T. Development of a biologically inspired hopping robot-" Kenken". InProceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292) 2002 May 11 (Vol. 4, pp. 3984-3991). IEEE.
6. Focchi M. Strategies to improve the impedance control performance of a quadruped robot. Genoa: Istituto Italiano di Tecnologia. 2013 Apr.
7. Ghosh R, Pratihar DK, Gupta S. Towards the optimal design of an uncemented acetabular component using genetic algorithms. Engineering Optimization. 2015 Dec 2;47(12):1587-601.
8. Saar KA, Giardina F, Iida F. Model-free design optimization of a hopping robot and its comparison with a human designer. IEEE Robotics and Automation Letters. 2018 Jan 22;3(2):1245-51.
9. Todorov, Emanuel and Erez, Tom and Tassa, Yuva. Mujoco: A physics engine for model-based control. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems.2012. 5026–5033.
10. Kraft D. A software package for sequential quadratic programming. Forschungsbericht- Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt. 1988.
11. Hansen N, Müller SD, Koumoutsakos P. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). Evolutionary computation. 2003 Mar;11(1):1-8.