An Energy Compensation Approach to Variable Stiffness Single Leg Jumping

1st Mohammadreza Gilak
Department of Electrical Engineering
Sharif University of Technology
Tehran, Iran
mohammadreza.gilak@ee.sharif.edu

2nd Mehdi Zarei
Department of Mechanical Engineering
Sharif University of Technology
Tehran, Iran
mehdi.zar@student.sharif.edu

3rd Mohammad Hossein Basiri
Fasta Robotics Inc.
Sharif Science and Technology Park
Tehran, Iran
basiri@fasta.technology

4th Saman Samiei
Department of Aerospace Engineering
Sharif University of Technology
Tehran, Iran
saman.samiei@ae.sharif.edu

5th Mohammad Mehdi Jalalmaab
Fasta Robotics Inc.
Sharif Science and Technology Park
Tehran, Iran
jalalmaab@fasta.technology

6th Behzad Ahi
Department of Electrical Engineering
Sharif University of Technology
Tehran, Iran
ahi@sharif.edu

***Abstract*—In the presented paper, we introduce a controller that enables stable and dynamic locomotion of a quadruped robot's single leg in the presence of friction. In this model, the bio-inspired template of a spring-loaded inverted pendulum is used to capture the essential features of jumping gaits observed in animals. In the flight phase, the generation of the foot-end trajectory, as well as the control of leg orientation using a Bézier curve, ensures smooth transitions. During the stance phase, the controller uses a varying-stiffness spring model to account for energy losses due to friction. Extensive simulations demonstrate the proposed framework's ability to compensate for energy losses and maintain the same jumping ability. They also show that the effects of friction can be minimized by iteratively updating the spring stiffness.**

***Index Terms*—spring-loaded inverted pendulum, single leg, test stand, quadrupedal robot, Bézier curve**

I. Introduction

Given the rapid growth of robotics technology and the expansion of its applications in diverse fields such as medicine, transportation, agriculture, and manufacturing industries, the need for intelligent, robust, and reliable robotic systems is increasingly felt [1]. Among the various types of robots, legged robots have gained a special position in research and practical applications due to their ability to move in uneven environments and to simulate the locomotion behavior of humans and animals [2], [3]. Some researchers have even used legged robots to explore the ocean beds [4]. The optimal performance of legged robots is directly tied to the quality of their control algorithms. The design and development of efficient control strategies not only enhance locomotion accuracy and speed, but also improve the system's reliability, stability, and adaptability. However, due to the nonlinear nature of their dynamics, the presence of uncertainties, and complex interactions with the environment, controlling legged robots remains one of the fundamental challenges in robotics [5], [6].

A fundamental advantage of legged robots over wheeled robots is their ability to traverse uneven terrain. This benefit is rooted in the decoupling between the robot's body and the environment. First, the movement of the robot's body becomes largely independent of the terrain's smoothness or roughness, as in many cases, the legs act as an active suspension system between the body and the ground. Second, this decoupling allows the legs to temporarily detach from the ground, enabling access to locations that would otherwise be completely unreachable under different conditions. But these advantages come at the cost of more complicated dynamics and dealing with hybrid systems in control design.

In recent years, several studies have sought to improve the control of legged robots. For instance, Tian et al. [7] employed fast quadratic programming processing to control a single joint. Some approaches also attempt to regulate the force exerted by the robot on the ground, aiming for more refined motion that contributes to the robot's longevity. This can be done by either measuring the foot force or estimating the force acting on the robot is estimated using a filter, which can then be incorporated into the controller design process [8]. However, these methods rely heavily on a precise model of the leg, which usually cannot be ensured in real-world applications.

To have the robotic leg interact with the environment, one approach is to incorporate compliant behavior in the actual leg. Lou et al. [9] proposed an energy-based rest length regulation for control of a compliant one-legged robot. Some robots can even switch between two modes of configurations, one of which is compliant [10]. Although physical compliance on a leg solves the problem of contacting the ground, adding elastic elements to a robot is often not possible and restricts the otherwise free motion of the robot.

Another method of dealing with the problem of contacting the ground in legged robots is incorporating a bio-inspired model called Spring-Loaded Inverted Pendulum (SLIP) model.

This concept will be explained in more detail in Sec. III. Huang et al. [11] combined the SLIP model with air trajectory planning for controlling a single leg with three degrees of freedom. The concept of SLIP can also be extended to three dimensional models [12], wheeled single legs [13] and even quadrupedal robots [14]. Building on this, in this paper, we propose a control algorithm based on SLIP modeling and air trajectory planning along with a new method to compute thrust force added during the stance phase. This new method enables the controller to compensate for the energy lost to friction during flight phase.

The rest of the paper is organized as follows: Sec. II reviews some basic concepts such as impedance control and Bézier curves. In Sec. III kinematics and dynamics of the single leg robot are provided along with the SLIP modeling used. In Sec. IV the control design procedure is presented. Simulations are conducted in Sec. V. Finally, Sec. VI concludes the paper with some suggestions for future works.

II. Preliminaries

A. Impedance Control

Legged robots must constantly control how they interact with the ground. Unlike wheeled robots, each leg repeatedly enters and leaves contact with environments that can be stiff, soft, compliant, uneven, or even moving. Leg control cannot rely purely on precise trajectory tracking, since the forces exerted by the ground on the robot are only partly predictable. Instead, the leg must balance the two objectives of allowing some deviation from planned motion so that unexpected contact forces do not destabilize the robot, and generating appropriate forces to support body weight, to attain locomotion, and to reject disturbances.

Impedance control provides a natural and physically intuitive framework for meeting these requirements. Impedance control, in the context of a single robotic leg, controls the dynamic relation between the motion of the leg and the ground reaction forces arising from stance and impact. Rather than commanding forces directly or attempting to track position rigidly, the controller shapes how the leg behaves mechanically, much like a tunable mass-spring-damper system.

This desired impedance behavior in a robot can be achieved in different manners. One solution is to employ springs and dampers in the structure of the robot [9]. Another option is to make the robot behave as an impedance using the control inputs. Any mechanical behavior can be imitated using only the control inputs.

Here, impedance is implemented in the robot behavior at joint level. This is due to the choice of motors, which incorporate field-oriented control themselves. Therefore, instead of using a rigid controller for reference tracking, a Proportional Derivative (PD) controller is used as

$$\tau = \tau_{ff} + K_p(q_{des} - q) + K_d(\dot{q}_{des} - \dot{q}), \tag{1}$$

where τ_{ff} is the feed-forward torque, K_p and K_d are the position and velocity gains, q_{des} and $\dot{q}_{des}$ are desired angle and angular velocity and q and $\dot{q}$ are the joint's angle and angular velocity, respectively.

B. Bézier curves

Bézier curves are an important class of functions used in computers, graphics, motion planning and even vehicle design. They have the feature that their derivatives of any order can also be specified in any point along the curve by using special points called control points. A Bézier curve of order n has $n+1$ control points.

The first-order Bézier curve is just a line between the two control points as

$$B_1(t) = (1-t)P_0 + tP_1, \ \ ,0 \le t \le 1, \tag{2}$$

where P_0 and P_1 are the control points. The second-order Bézier curve is defined like

$$B_2(t) = (1-t)B_1^{P_0,P_1} + tB_1^{P_1,P_2}, \ \ ,0 \le t \le 1, \tag{3}$$

where this time P_0, P_1 and P_2 are the three control points. Higher order Bézier curves are defined recursively in a similar manner. In the context of this paper, 5th order Bézier curves are used to ensure smooth foot trajectory during flight phase.

III. Modelling

A. Kinematics

The single leg investigated in this paper is a planar robot with two degrees of freedom. It consists of a hip joint mounted on a vertical slider and a knee joint. The kinematics of the robot are shown in Fig. 1, in which hip and knee angles are represented by q_1 and q_2, respectively. The position of the end of the foot with respect to the hip can be written as

$$x = -l_1 \cos(q_1) - l_2 \cos(q_1 + q_2), \tag{4}$$

$$y = -l_1 \sin(q_1) - l_2 \sin(q_1 + q_2). \tag{5}$$

A fundamental tool in robotics is the Jacobian matrix since it relates joint velocities to end-effector velocities. The Jacobian

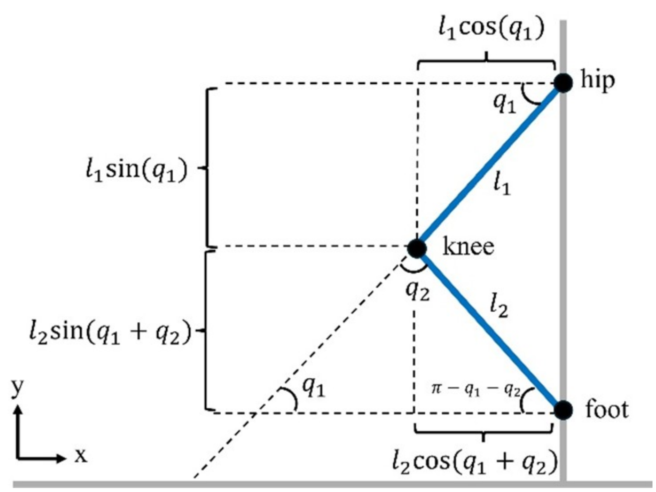

Fig. 1. Kinematics of the Robotic Leg.

is essential for tasks like control, inverse kinematics, and force mapping. Here, we consider the foot as the end effector, and since its orientation is not of interest, we can write

$$\dot{x} = l_1 \dot{q}_1 \sin(q_1) + l_2(\dot{q}_1 + \dot{q}_2) \sin(q_1 + q_2), \tag{6}$$

$$\dot{y} = -l_1 \dot{q}_1 \cos(q_1) - l_2(\dot{q}_1 + \dot{q}_2) \cos(q_1 + q_2). \tag{7}$$

The Jacobian is the matrix of partial derivatives of the end-effector position with respect to the joint angles.

$$J = \begin{bmatrix} l_1 \sin(q_1) + l_2 \sin(q_1 + q_2) & l_2 \sin(q_1 + q_2) \\ -l_1 \cos(q_1) - l_2 \cos(q_1 + q_2) & -l_2 \cos(q_1 + q_2) \end{bmatrix} \tag{8}$$

B. Dynamics

Two main methods exist for writing the dynamics of a robot. The first one is using Newtonian dynamics, which is prevalent in manipulator control literature. The second method is using the Lagrangian formulation of the problem. Although the Newtonian approach is simpler for most systems, for systems that interact with the environment, the Lagrangian method is preferred, because it is much simpler to consider constraints.

The Lagrangian method describes the way energy is transferred in the system. The procedure involves finding the kinetic and potential energy of the system, forming the Lagrangian and adding constraints. Here, writing every term of kinetic and potential energy is skipped. Instead, it is assumed that $K(q_1, q_2, \dot{q}_1, \dot{q}_2)$ and $P(q_1, q_2)$ are the kinetic and potential energy of the system, respectively. The Lagrangian can then be written as

$$\mathcal{L} = K(q_1, q_2, \dot{q}_1, \dot{q}_2) + P(q_1, q_2) + Fh \tag{9}$$

where h is the height of the end of the foot. This constraint ensures that the foot cannot penetrate the ground or $h \geq 0$. Note that although F is the free constraint variable in the Lagrangian, it has an intuitive meaning behind it, in the sense that it represents the ground reaction force acting on the foot.

The dynamics of the robot can now be written as

$$\frac{\partial \mathcal{L}}{\partial q_1} - \frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \mathcal{L}}{\partial \dot{q}_1} = 0, \tag{10}$$

$$\frac{\partial \mathcal{L}}{\partial q_2} - \frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \mathcal{L}}{\partial \dot{q}_2} = 0, \tag{11}$$

$$\frac{\partial \mathcal{L}}{\partial h} - \frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \mathcal{L}}{\partial \dot{h}} = 0, \tag{12}$$

From this, $\ddot{q}_1$, $\ddot{q}_2$ and $\ddot{h}$ or F are derived. The resulting dynamics can be simulated to obtain a very simple model of the system.

C. Spring-Loaded Inverted Pendulum Modelling

In the field of legged robotics, it is customary to inspire models from the nature. SLIP is a simple bio-inspired model that describes the running gait of many quadruped animals.

Basically in SLIP modeling, the whole leg is represented by a mass-less spring attached to a mass. The behavior of this structure matches the behavior observed in running gates of many animals. It is important to note here that a mass-spring system has hybrid dynamics. The mass moves under the force of gravity during flight phase. Then it makes contact with the ground and enters the compression phase.

In this phase, the vertical velocity of the center of mass of the leg is negative, the spring is being compressed and energy is being stored in the spring. When the spring starts expanding, the system enters thrust phase. Here, the energy stored in the spring as potential energy is released as kinetic energy which makes the leg jump. In the absence of outside forces, compression and thrust phases dynamics are the same.

In the case of an animal, the horizontal speed of running can be changed by modifying the touchdown angle, which is the angle of the hip at the time the leg makes contact with the ground. For the single leg considered in this paper, which is mounted on a stand and cannot move, desired horizontal speed is zero and the touchdown angle is always 0°. The height of the apex, which is the highest point the leg can achieve, depends on the stiffness and nominal length of the spring.

IV. Controller Design

The controller employed here bears close resemblance to that of Huang et al. [11], whose approach has become widespread within the legged robotics literature. A four-state finite state machine governs the motion. The first state, compression, corresponds to the phase in which the leg behaves like a spring being compressed and accumulating elastic energy. The second state, thrust, follows as the leg begins to extend, driven by the force stored during compression. The third state, swing, is characterized by the foot-end tracking a trajectory generated via Bézier curves. The final state, landing, occurs once the Bézier curve timing has elapsed and the virtual spring has returned to its nominal length, preparing the leg for touchdown. The finite state machine is shown in Fig. 2. During swing phase, the foot follows a 5th order Bézier curve, whose six control points are defined as

$$P = \begin{bmatrix} x_{lo} & x_{lo} & x_{lo} & x_{lo} & x_{td} & x_{td} \\ y_{lo} & y_{lo} & y_{lo} + H & y_{lo} + H & y_{td} & y_{td} \end{bmatrix}, \tag{13}$$

where (x_lo, y_lo) and (x_td, y_td) are lift-off and touchdown coordinates, respectively and H is a tunable variable, allowing the robot to lift its leg higher in case it needs to jump higher. The Bézier curve generated with the control points in (13) defines a desired trajectory for the foot-end. This trajectory is transformed into joint angles and angular velocities using the inverse kinematics of the leg. The controller in (1) is then used to follow this trajectory.

The generated Bézier curve also includes time, meaning it is a trajectory, not a path. This means it might end before the foot has reached the ground. This is where the landing phase comes in. In the landing phase, the leg holds its configuration, so that the spring is at nominal length and the touchdown angle is preserved. The landing phase also utilizes (1) to keep the desired pose.

When the foot makes contact with the ground, it enters the compression phase. In this phase, a virtual spring system

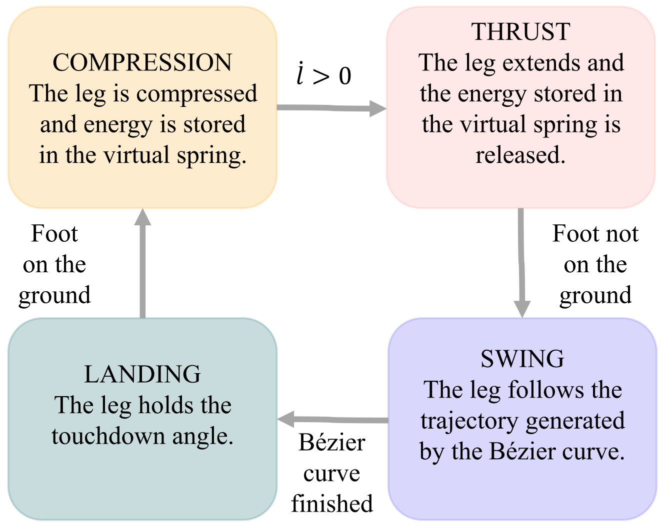

Fig. 2. Controller finite state machine.

is considered and the force produced by this structure is calculated as

$$F_{sd} = K_s(l - l_{des}), \tag{14}$$

where l is the length of the virtual spring considered at the foot, l_{des} is the length of the virtual spring at rest and K_s is the spring coefficient. This force slows down the descent of the leg, making the contact less harsh.

When the virtual spring starts extending back to its nominal length, the leg enters thrust mode. In thrust phase, the same force is applied as the compression phase, with the difference that this time the spring coefficient is different. The new coefficient is computed as

$$K'_s = K_s + \Delta K, \tag{15}$$

in which ΔK is calculated such that enough energy is added to the system so that the system can reach its desired apex height. To this end, it is assumed that all the energy added to the system in this phase is converted into gravitational potential energy.

$$\frac{1}{2}\Delta K(l_{\min} - l_{des})^2 = mg\Delta h, \tag{16}$$

where Δh equals $h_{des} - h$, which is the error of height. It should be noted that when the leg is the most compressed and the length of the virtual leg is equal to $l_{\min}$, no potential energy is stored in the system and all the energy of the system is that stored in the spring. Also, the effects of slider and joint frictions can then be mitigated by updating the spring constant at every sample, because every time that the leg doesn't reach the desired height due to friction, the spring constant changes to compensate for that.

From (16), ΔK can be computed as

$$\Delta K = \frac{2mg\Delta h}{(l_{\min} - l_{des})^2} \tag{17}$$

Fig. 3. The robotic leg in the Issac environment.

While a complete Poincare map analysis is out of the scope of this paper, extensive simulations have been done to show the effectiveness of the method.

V. Simulation Analysis

In this section, the proposed scheme is thoroughly investigated. The simulations are conducted in Isaac Environment due to its precision and accurate physics. The robotic leg in different phases can be seen in Fig 3. Two sets of simulations were carried out. In the first experiment, friction was taken into consideration to accurately model the real-world system, but it was not compensated. In the second experiment, the proposed approach was used to negate the energy loss caused by friction by adding energy in the thrust phase.

The experimental results are shown in Fig. 4, for which hip height is used as a measure of jumping ability; because it is well established in this context as an appropriate indicator of robotic agility and performance. In the frictionless case, the leg attains its maximum jump height. If friction is present, the leg cannot support the same height of jumping; its energy is progressively dissipated over a few cycles and the motion eventually ceases completely. In contrast, when the proposed method is applied, despite continuous energy losses due to friction, the robot is able to sustain repeated jumping by means of energy injection during the thrust phase. It can be seen that the desired jumping height is achieved in a few cycles.

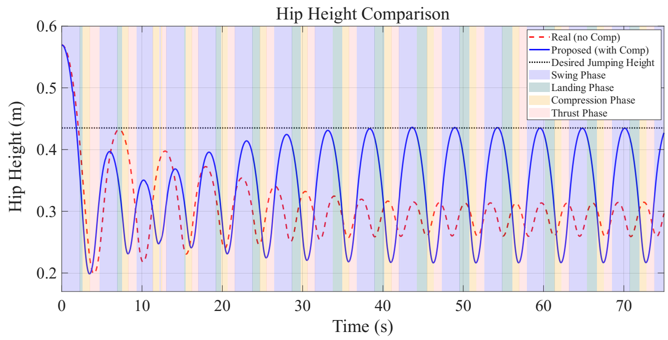

Fig. 4. Hip height compared for the two experiments.

The spring constant is also presented in fig. 5. The spring constant first changes dramatically to achieve the desired height and then settles to an appropriate value to compensate the energy losses in the system.

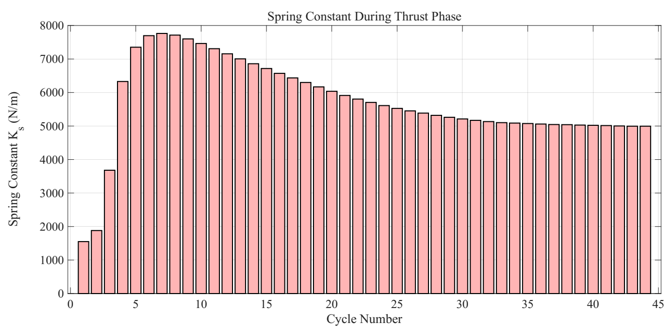

Fig. 5. Spring constant changes in different cycles' thrust phase.

VI. Conclusion

This paper presented a novel energy conservation-based method for reaching a desired height in a single leg robotic platform. The method consisted of a finite state machine controller with a variable spring stiffness during the thrust phase. Extensive simulations confirmed the effectiveness of the proposed method. Future works can be done on deriving a more rigorous proof of stability or an additional term for handling friction.

Acknowledgment

No external funding was received for this work. The authors declare no conflict of interest.

References

[1] I. Al-Tameemi and O. Amanuel, "Bipedal robots: A systematic review of dynamical models, balance control strategies, and locomotion methods," *Journal of Robotics and Control (JRC)*, vol. 6, no. 3, pp. 1240–1254, 2025.

[2] J. Reher and A. D. Ames, "Dynamic walking: Toward agile and efficient bipedal robots," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 4, no. 1, pp. 535–572, 2021.

[3] Y. Tong, H. Liu, and Z. Zhang, "Advancements in humanoid robots: A comprehensive review and future prospects," *IEEE/CAA Journal of Automatica Sinica*, vol. 11, no. 2, pp. 301–328, 2024.

[4] H. Yang, Y. Din, Q. Liu, and H. Zhang, "Control of a single-leg robot based on closed-loop pd-type iterative learning and neural network," *2024 International Conference on Advanced Control Systems and Automation Technologies (ACSAT)*, pp. 170–177, 2024.

[5] F. N. Martins, "Smart robotics for automation," *Sensors*, vol. 24, no. 12, p. 3900, 2024.

[6] M. Sombolestan and Q. Nguyen, "Adaptive-force-based control of dynamic legged locomotion over uneven terrain," *IEEE Transactions on Robotics*, vol. 40, pp. 2462–2477, 2024.

[7] D. Tian, J. Gao, C. Liu, and X. Shi, "Simulation of upward jump control for one-legged robot based on qp optimization," *Sensors*, vol. 21, no. 5, p. 1893, 2021.

[8] Y. Fan, Z. Pei, and Z. Tang, "Variable impedance control for a single leg of a quadruped robot based on contact force estimation," *International Journal of Control, Automation and Systems*, vol. 22, no. 4, pp. 1360–1370, 2024.

[9] G. Luo, R. Du, S. Song, H. Yuan, Z. Huang, H. Zhou, and J. Gu, "Stable and fast planar jumping control design for a compliant one-legged robot," *Micromachines*, vol. 13, no. 8, p. 1261, 2022.

[10] F. Nan, H. Kolvenbach, and M. Hutter, "A reconfigurable leg for walking robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1308–1315, 2021.

[11] S. Huang and X. Zhang, "Controlling a one-legged robot to clear obstacles by combining the slip model with air trajectory planning," *Biomimetics*, vol. 8, no. 1, p. 66, 2023.

[12] B. Han, H. Yi, Z. Xu, X. Yang, and X. Luo, "3d-slip model based dynamic stability strategy for legged robots with impact disturbance rejection," *Scientific Reports*, vol. 12, no. 1, p. 5892, 2022.

[13] J. Gao, H. Jin, L. Gao, Y. Zhu, J. Zhao, and H. Cai, "Jump control based on nonlinear wheel-spring-loaded inverted pendulum model: Validation of a wheeled-bipedal robot with single-degree-of-freedom legs," *Biomimetics*, vol. 10, no. 4, p. 246, 2025.

[14] M. S. U. Hassan, D. Vasquez, H. Asif, and C. Hubicki, "Stable and robust slip model control via energy conservation-based feedback cancellation for quadrupedal applications," *arXiv preprint arXiv:2511.05402*, 2025.