Bio-Inspired Leg

Design and characterisation of a bio-inspired multi-segmented leg consisting of spring and tendon

Semester project

Maïk Guihard

EPFL
Ecole Polytechnique Fédérale de Lausanne

Bio-Inspired Leg

Design and characterisation of a bio-inspired multi-segmented leg consisting of spring and tendon

by

Maïk Guihard

284922

Instructor:	Prof. Auke Ijspeert
Teaching Assistants:	Milad Shafiee, Guillaume Bellegarda
Project Duration:	February 2022 - June 2022
Faculty:	STI, EPFL

Cover:	The Rolex Learning Center at EPFL (Modified)
Style:	EPFL Report Style, with modifications by Batuhan Faik Derinbay

EPFL

Contents

1

Introduction

Most felines and some land avians have very impressive running and jumping abilities. They can jump several times their body heights and run at very high speeds, as well as perform extremely highly agile movements. This is in part due to several elastic muscle-tendon complexes which they possess, allowing them to passively store energy during stance and release it to enhance their movements. Such structures are found in humans too, which help us walk through increased ankle torque at pushoff.

In the field of legged robotics, incorporating such structures can greatly increase the performance of mobile robots. Currently, most commercially available quadrupeds use high-torque proprioceptive actuators with complicated control policies. The two principal industrial quadrupeds, used for inspection, are the SPOT [6] and the ANYmal [1]. They do not incorporate passive elastic elements as their control policies are based on very accurate position and velocity control. For that reason, they are not able to perform very agile maneuvers or run at high speeds. Also, they require constant torque output to stand up which is energetically costly.

In order to gain those capabilities, incorporating parallel or series elastic elements seems like the logical bio-inspired approach. Because series elastic actuators introduce uncertainty on the position, parallel elastic actuation seems to be the most promising solution. This is the next step in legged robotics to advance to highly agile robots. This agility is needed to perform tasks currently unavailable to commercial quadrupeds, like jumping over large obstacles or running at useful speeds.

Biological muscles and tendons can be imitated in engineering by using springs and rigid or elastic cables. When placed intelligently and combined with pulleys and cams, they passively generate torques similar to those observed in nature. In order to retain accurate position control while incorporating springs and tendons, a good structure is the pantograph leg. The end-effector's position is perfectly defined by the hip and knee motors' position, so the elastic elements do not introduce uncertainty.

Some research has been conducted to explore the integration of spring in robotic legs. Sato et.al. [4] and Wang et.al.[7] highlighted that the two most promising muscle-tendon complexes found in nature are the gastrocnemius muscle with the achille's tendon, and the rectus femoris with the patellar tendon, found in cats and many other animals. The former creates an extension torque at the ankle, while the latter extends the knee. These two structures store energy during stance and release it during pushoff to jump or run better. In a pantograph structure, the knee and ankle joints are directly linked, so it is interesting to compare the performance of the two structures. Badri-Spröwitz et.al. [3] developed an avian-inspired leg that imitates the same muscle-tendon complexes to increase energy efficiency.

Another advantage of parallel elastic elements is that they can support the weight of the robot without requiring actuation. This can allow quadrupeds to only use energy for movement and passively stand up. Even for the current industry applications of such robots, where they stand most of the time and walk slowly, alleviating the motors of some of the work can greatly increase their energy efficiency and autonomy.

A second interesting structure in biology is the toe. In some animals like horses and birds, it acts as a sort of clutch for elastic structures. When the toe is in contact with the ground, it tensions ligaments behind the foot that contribute to body weight support. On a robotic leg, by connecting the aforementioned springs and tendons to the toe in a manner similar to biology, it is possible to realize a clutch mechanism that releases all tension in the tendon during flight phase. As shown in Fig. 1.1,

when the toe is not in contact with the ground, it can rotate freely which negates all tension in the system. By adding this mechanism to a parallel actuated robotic leg, it can avoid wasting energy in flight phase because the motors do not have to move against the spring to reposition the leg in the air before landing.

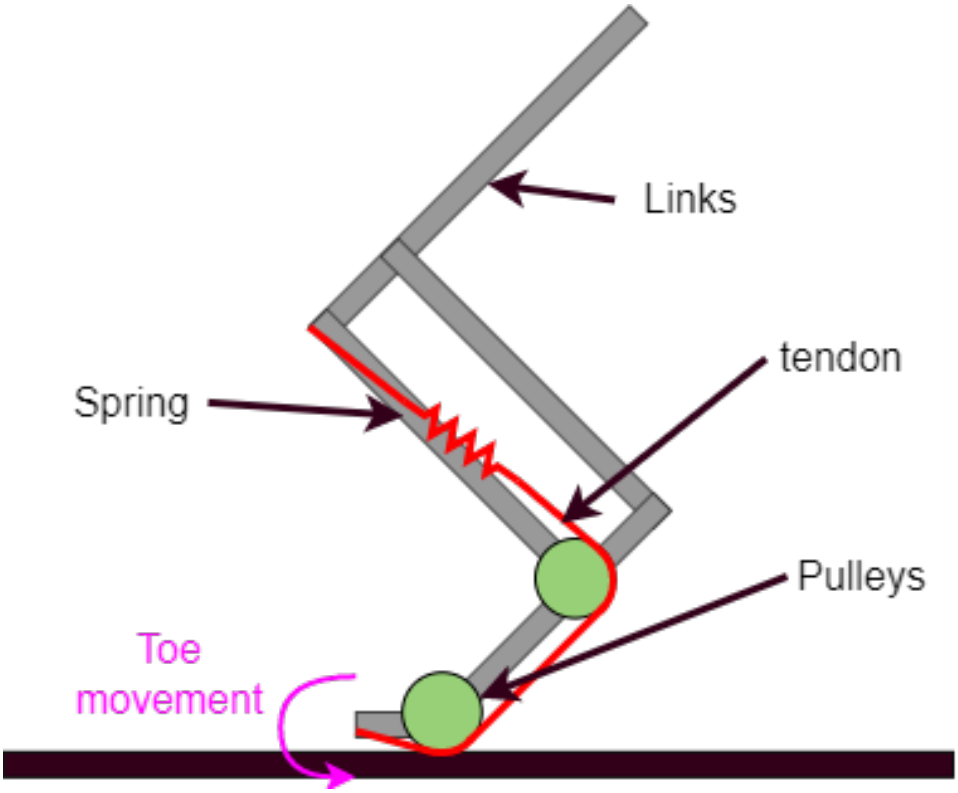


Figure 1.1: Contact-activated tendon tensioning with a toe

The goal of this project was to design and characterize a robotic leg incorporating hybrid (passive and active) actuation with engagement and dis-engagement using a toe, to have more efficient leg locomotion for jumping and bounding. The beginning of this project was done in collaboration with Erfan Etesami, who implemented force-feedback control on the proposed leg design.

2

Leg design

This project aims at incorporating springs and tendons to imitate the structures found in biology. Therefore it makes sense to design a leg with a shape and a number of links similar to what is found in animals. For that reason, a 3-link design was chosen, representing the thigh, calf and foot. On top of that, the addition of a toe allows to engage and dis-engage the spring mechanisms as explained in section 2.3.

The advantage of the pantograph structure is that it reduces the amount of actuators necessary in the system. Indeed, the two parallel links of the calf have the same length and are completely rigid, meaning that the angle at the knee and heel joints are directly equal, and the foot is always parallel to the thigh. Therefore it is sufficient to have an actuator at the hip and one at the knee to control all links down to the foot. The toe is not actuated but its position is constrained by springs and tendons (see section 2.3).

2.1. Dimensioning

The leg dimensioning is heavily based on the MIT Cheetah [8] and Solo [5] robots, which are both quadrupeds. In order to simplify the process of building and testing a prototype after this project, the leg was designed to accommodate the Solo, which is an open source robot design. The 3-link pantograph leg is found in the MIT Cheetah, so it was used as a reference for scale between links. The number and sizes of links for the MIT cheetah, Solo, and this project's design are shown on Fig. 2.1.

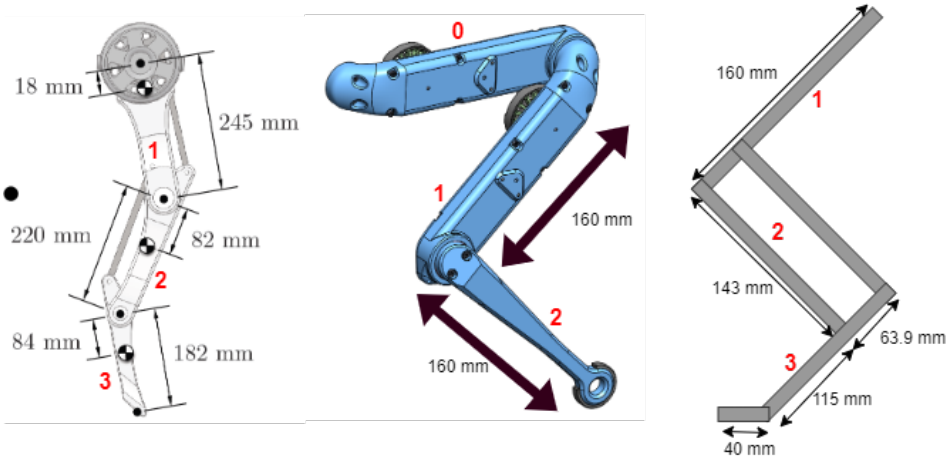


Figure 2.1: Sizes of links for (left to right) the MIT cheetah [8], Solo robot [5] and proposed design

In the Solo, link 0 is fixed to the base and contains actuation for the hip. Link 1 contains actuation for the knee joint, and it is free. Because extensive testing was previously done on the actuator parts

of the design, it was decided to keep the first two links of the Solo as they are, and design the other links accordingly. They were dimensioned as a scaling of the MIT Cheetah, by taking the solo's first moving link's size as a reference. The resulting design has a maximum height of 40cm and a resting height of 30cm.

2.2. Mechanical model

After being dimensioned, a simple model of the leg was built in Solidworks (Figure 2.2), incorporating the actuator module of the Solo. All joints in the model are prismatic joints. Once put together in an assembly, the system was exported to the Unified Robotics Description Format (URDF) format, both for a single leg and for a whole quadruped. After some minor manual modifications to the file (joint limits, masses), it was ready for simulation. A mass corresponding to $\frac{1}{4}$ of the Solo's body weight was added on top in order to simulate what one leg has to lift.

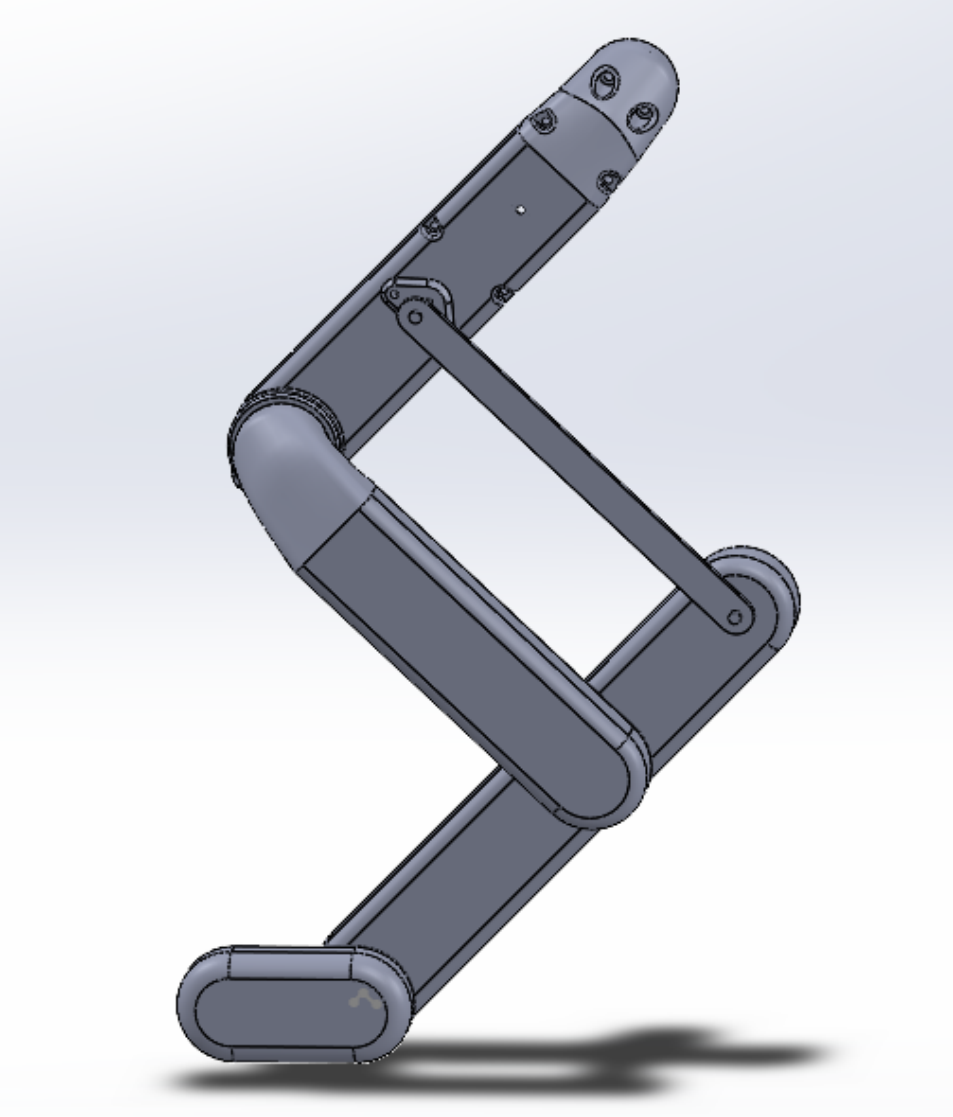


Figure 2.2: Simple CAD of the leg for simulation

2.3. Spring configurations

The objective is to add elasticity in the system so that the leg stores elastic energy when touching the ground, and releases it for jumping. The elastic elements can be placed either in series with the actuation, on in parallel. When placed in series, some accuracy on the position is lost, which can be

a problem for quadrupeds as they need precise movements for stability. For that reason, this project focuses on parallel elastic actuation.

When looking at biology, there are two main muscle-tendon complexes that store energy for jumping. First are the gastrocnemius muscle along with the achille's tendon, which exert force at the ankle to extend the foot. Second are the rectus femoris with the patellar tendon, which exert an extension force at the knee. Both muscle-tendon complexes are found in several classes of animals like avians, felines and humans. They both exert a torque by being attached to the bones away from the joints' center of rotation, which can be imitated in robotics with pulleys and cams. In the proposed leg design, when the leg extends or contracts, the movements and rotations can be exploited to generate torques with conveniently placed springs and tendons.

One big drawback of parallel elastic actuation is that the leg is not able to be moved without applying torque. More specifically, during the swing phase, the mechanism keeps the leg in a fully extended position. Therefore any wanted movements during swing phase, for example for positioning the leg before landing, will go against the spring and the motors will waste some energy in the process. To avoid that, there needs to be a dis-engaging mechanism that removes tension in the spring during flight phase. This can be accomplished by leveraging a toe and by attaching a tendon below it. When the leg is in stance phase, the toe is blocked by the ground and can not rotate, which allows the tendon to be tensioned (see Fig. 1.1). However during swing phase, the toe is not blocked any more and can rotate freely backwards, which removes tension from the tendon and permits the leg to be moved freely without being restrained by the string, thus saving energy.

Three spring configurations were tested, shown in Fig. 2.3. Configuration A imitates the gastrocnemius and achille's tendon, with the tendon attached to the toe for engagement and dis-engagement. Configuration B imitates the rectus femoris with patellar tendon. However due to its placement, it is not connected to the foot and therefore can not take advantage of the toe mechanism. For that reason, configuration C is a combination of A and B, with a long tendon stretching from the thigh to the toe and imitating both tendons as if they were connected.

For the three configurations, the resting position for the toe is chosen to be colinear with the foot. This ensures that the system engages when the foot touches the ground, as it will push the toe up. Because of the tendon, the application of force is unidirectional, meaning if the toe is retracted to the back, these configurations will not push it forward to the rest position. This is done with a second spring and tendon, as explained in section 2.3.4.

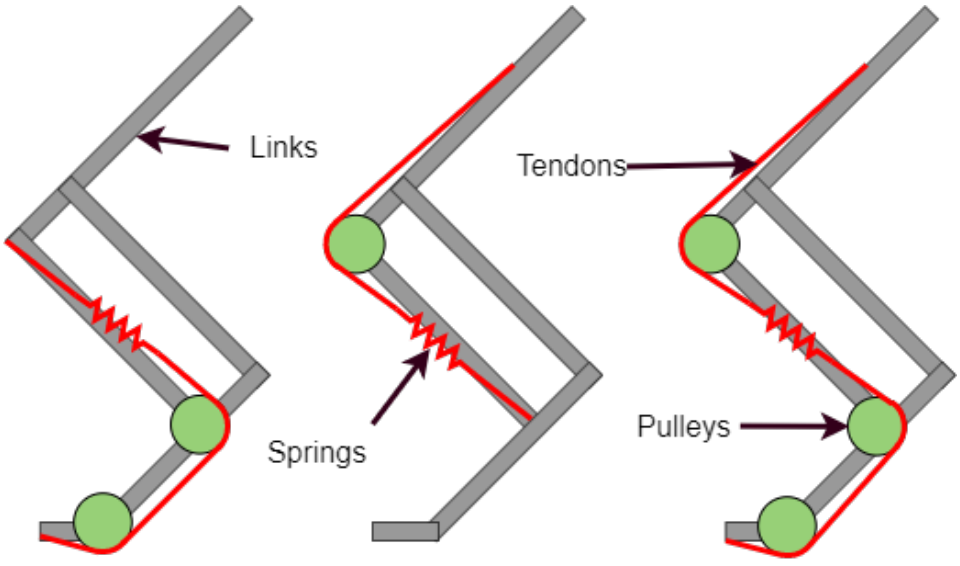


Figure 2.3: Spring configurations A, B and C

2.3.1. Configuration A

In this configuration, shown in Fig. 2.4, the spring and tendon imitate the gastrocnemius and achille's tendon. It applies a torque at the ankle and toe joints. These torques can be calculated as follows:

$$\Delta x = \theta_1 \cdot \frac{r_1}{r_2} \cdot r_3 + \theta_2 \cdot r_3 = (\theta_1 \cdot \frac{r_1}{r_2} + \theta_2) \cdot r_3 \tag{2.1}$$

$$F_1 = \Delta x \cdot k \tag{2.2}$$

$$F_2 = \Delta x \cdot k \cdot \frac{r_3}{r_2} \tag{2.3}$$

$$T_{ankle} = F_1 \cdot r_3 = k \cdot r_3^2 \cdot (\theta_1 \cdot \frac{r_1}{r_2} + \theta_2) \tag{2.4}$$

$$T_{toe} = F_2 \cdot r_1 = k \cdot \frac{r_3^2 \cdot r_1}{r_2} \cdot (\theta_1 \cdot \frac{r_1}{r_2} + \theta_2) = \frac{r_1}{r_2} \cdot T_{knee} \tag{2.5}$$

where Δx is the spring compression/extension, r_1, r_2 and r_3 are the pulley radii, and angles and torques are set as in Fig. 2.4. The pulleys r_2 and r_3 are attached together and changing their radii allows to tune the distribution of forces and torques along the tendons.

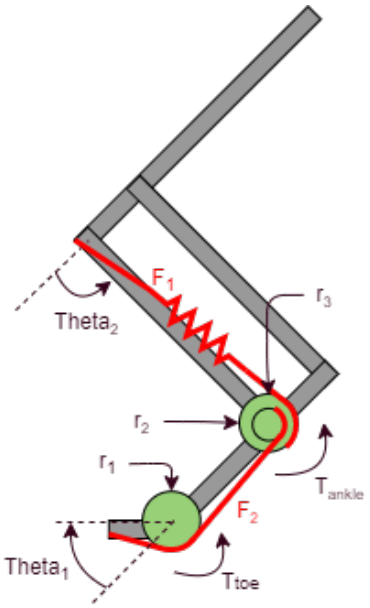


Figure 2.4: Details for torque calculations on configuration A

2.3.2. Configuration B

In this configuration, shown in Fig. 2.5, the spring and tendon imitate the rectus femoris and patellar tendon. It applies a torque at the knee joint, which can be calculated simply as follows:

$$\Delta x = \theta_1 \cdot r_3 \tag{2.6}$$

$$F_1 = \Delta x \cdot k \tag{2.7}$$

$$T_{knee} = F_1 \cdot r_3 = \theta_1 \cdot k \cdot r_3^2 \tag{2.8}$$

where Δx is the spring compression/extension, r_1 is the pulley radius and angles and torques are set as in Fig. 2.5. It should be noted that this configuration does not connect to the toe and has no dis-engagement capabilities.

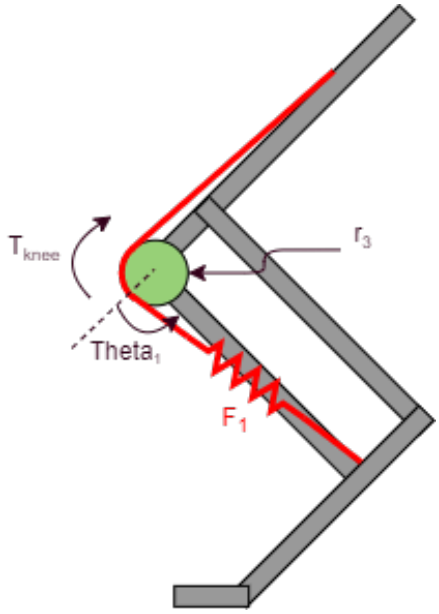


Figure 2.5: Details for torque calculations on configuration B

2.3.3. Configuration C

In this configuration, shown in Fig. 2.6, the spring and tendon connect the configuration B to the toe, effectively acting as a combination of both previous configurations. It applies a torque at the knee, ankle and toe joints. In a pantograph configuration, applying a torque at the ankle or knee is equivalent, therefore both knee and ankle torques are compressed into one. These torques can be calculated as follows:

$$\Delta x = \theta_1 \cdot \frac{r_1}{r_2} \cdot r_3 + 2 \cdot \theta_2 \cdot r_3 = (\theta_1 \cdot \frac{r_1}{r_2} + 2 \cdot \theta_2) \cdot r_3 \tag{2.9}$$

$$F_1 = \Delta x \cdot k \tag{2.10}$$

$$F_2 = \Delta x \cdot k \cdot \frac{r_3}{r_2} \tag{2.11}$$

$$T_{knee} = 2 \cdot F_1 \cdot r_3 = 2 \cdot k \cdot r_3^2 \cdot (\theta_1 \cdot \frac{r_1}{r_2} + 2 \cdot \theta_2) \tag{2.12}$$

$$T_{toe} = F_2 \cdot r_1 = k \cdot \frac{r_3^2 \cdot r_1}{r_2} \cdot (\theta_1 \cdot \frac{r_1}{r_2} + 2 \cdot \theta_2) = \frac{r_1}{2 \cdot r_2} \cdot T_{knee} \tag{2.13}$$

where Δx is the spring compression/extension, r_1, r_2 and r_3 are the pulley radii (pulleys 2 and 3 are combined), and angles and torques are set as in Fig. 2.6.

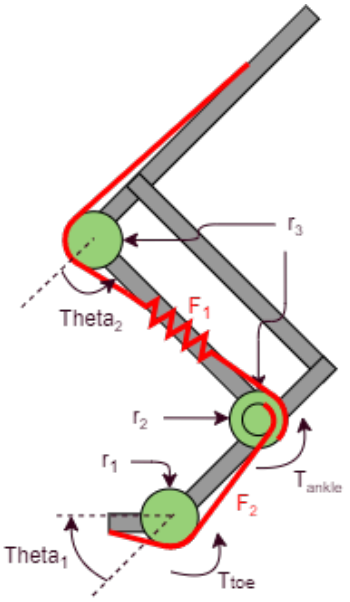


Figure 2.6: Details for torque calculations on configuration C

2.3.4. Toe restoring mechanism

When in the air, the toe is free to rotate, which is good for dis-engagement of the spring mechanism. However, it is important and necessary that it comes back to a reasonable position for landing, otherwise the robot might step on its toe, the mechanism would not be engaged and the movement would be disturbed. In order to prevent that, a solution is to add a small spring and tendon to bring the toe back to a good position, as shown on Fig. 2.7. This way, when the leg is completely extended for landing, the spring configurations are completely slack and the toe is brought back by this added part, which faces no resistance. Of course it opposes motor movement during swing, but the spring constant can be as arbitrarily small as wanted, so the energy lost is very minimal. The toe's resting position for this spring-tendon is chosen to be the same as for configurations A,b or C, namely a toe colinear to the foot. Again, this system is uni-directional, meaning it will pull the toe to the rest position when it is rotated back, but no force will be applied during stance as the toe will be rotated the other way by the ground contact.

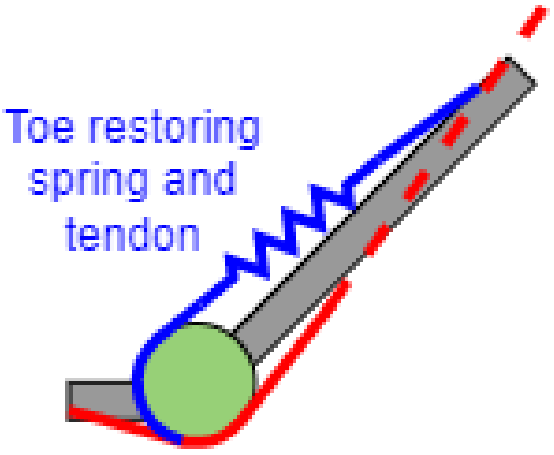


Figure 2.7: Toe restoring mechanism

Spring and pulley sizing

It was decided that the spring and tendon complexes should be able to support completely the robot for a fall of 3 body heights, which is 1.2m. For that, the necessary spring constant can be calculated

with a mechanical energy conservation calculation:

$$mgh = \frac{1}{2}k\Delta x_{max}^2 \tag{2.14}$$

with $m = 0.5kg$ (one quarter of the total robot weight, which is the weight that each leg needs to lift), $h = 1.2m$, and Δx_{max} is the maximum spring displacement. Because of the geometry of the system and the sizes of each link, the maximum range for the knee and ankle joints is 110°, and the toe joint moves by 90°between the unloaded and fully loaded leg. From this, the required spring constant can be calculated using equation 2.14 along with 2.1, 2.6 and 2.9 respectively. Using $r_1 = 2cm$ and $r_2 = r_3 = 4cm$ as a relatively random baseline before optimizing, the minimum required spring constants are shown in Table 2.1.

configuration	A	B	C
k_{min}	1005	1995	343

Table 2.1: Minimum spring constant for each configuration

3

Simulation environment

3.1. Constraints

In order to simulate the system, the URDF model was imported in python, using the simulator Pybullet. The simulation was done at 1000Hz for the control loop. In the resting position, all angles of the leg are either 90°or 45°, like in Fig. 3.1. Because URDF files only allow rigid elements, it is not possible to simulate the spring and tendons physically. Therefore the model is only comprised of the different links, and during simulation, the torques at each joint are calculated according to equations from section 2.3 for the chosen spring configuration, and are applied to the system as if by the motors. Also, URDF files are arranged in a tree-like structure, which does not allow to have loops. However in the pantograph desin, there is a loop. To circumvent that problem, three of the four joints in the pantograph are taken from URDF and the fourth one is added manually in pybullet, by introducing a point-to-point constraint at the right place.

In order to prevent the leg from falling forward or to the side, a second constraint is introduced between the base of the leg and the environment. The base is a volume-less object placed at the hip, containing $\frac{1}{4}$ of the mass of the robot's body to simulate the weight it would have to lift. It is constrained on a vertical axis with a prismatic joint.

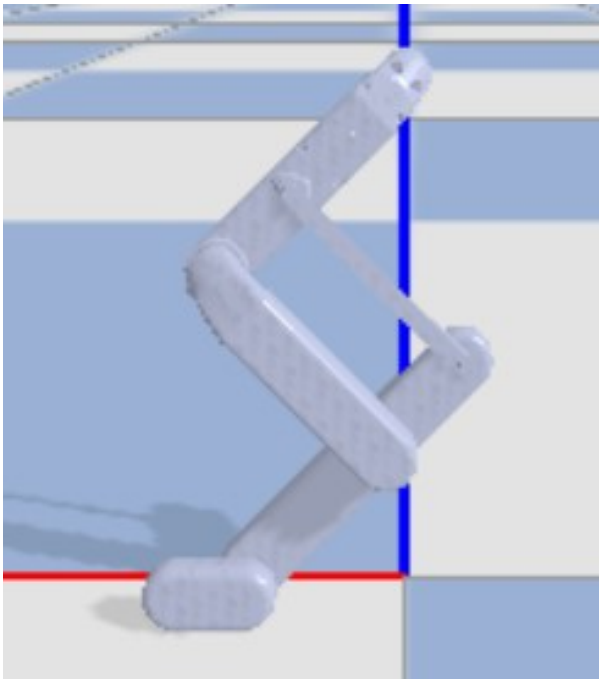


Figure 3.1: Base position of the leg in the simulator

3.2. Control policy

The objective is to explore how much energy the different spring configurations can save, how they change the jumping height and the impact of control delay. For that, the control is separated in three parts: jumping, waiting and landing. During jumping, the leg pushes down in order to jump, using a cartesian PD controller. The objective speed at takeoff is 4.2 m/s because it would lead to a height reached of 3 body heights (using $mgh = \frac{1}{2}mv^2$). The PD gains are tuned by hand for the best result. Once the leg has taken off, it gets in waiting mode. It does not try to apply a force, instead it places its end-effector (the base of the toe) vertically below the hip in order to be ready for landing. Once the toe touches the ground, the legs enters landing mode. Here it still maintains the toe below the hip to keep its balance, but no upwards force is applied still. In this mode the spring stores energy and slows and stops the leg before it crashes to the ground. When it reaches its lowest point and stops thanks to the spring and tendon, jumping mode is activated again and a new jump is initiated.

All throughout the simulation, at each time step, the motor torques are computed according to section 3.3 from the force command given by the PD controller. They are saturated to $\pm$ 2.7Nm because that is the maximum torque that the actuators from the Solo robot can produce. Then the spring torques are calculated from section 2.3 and are added to the motor torques, to be applied to the system.

3.3. Force and Jacobian calculations

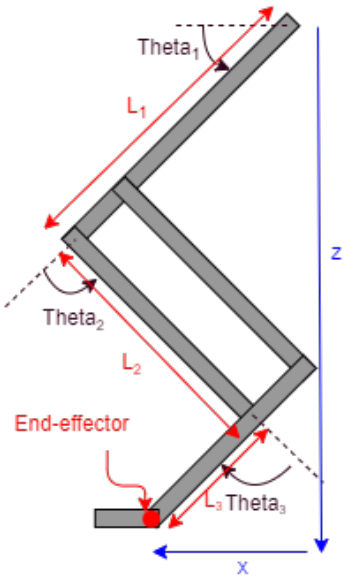


Figure 3.2: Diagram of a leg for Jacobian calculation

In order to transform the end-effector's force command into a torque command, the Jacobian of the system is needed. First, the end-effector's position is expressed as a function of the current joint angles, according to Fig. 3.2. It should be noted that $\theta_3 = \theta_2$ at all times because of the pantograph structure, giving:

$$x = l_1 \cdot cos(\theta_1) + l_2 \cdot cos(\theta_1 + \theta_2) + l_3 \cdot cos(\theta_1 + \theta_2 - \theta_3) = (l_1 + l_3) \cdot cos(\theta_1) + l_2 \cdot cos(\theta_1 + \theta_2) \quad (3.1)$$

$$z = l_1 \cdot sin(\theta_1) + l_2 \cdot sin(\theta_1 + \theta_2) + l_3 \cdot sin(\theta_1 + \theta_2 - \theta_3) = (l_1 + l_3) \cdot sin(\theta_1) + l_2 \cdot sin(\theta_1 + \theta_2) \quad (3.2)$$

As the end-effector's position has two coordinates (x and z) which depend on the positions of two joints (θ_1 and θ_2), the Jacobian is a 2x2 matrix:

$$J = \begin{pmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{pmatrix} \quad (3.3)$$

with:

$$J_{11} = \frac{\partial x}{\partial \theta_1} = -(l_1 + l_3) \cdot sin(\theta_1) - l_2 \cdot sin(\theta_1 + \theta_2) \tag{3.4}$$

$$J_{12} = \frac{\partial x}{\partial \theta_2} = -l_2 \cdot sin(\theta_1 + \theta_2) \tag{3.5}$$

$$J_{21} = \frac{\partial z}{\partial \theta_1} = (l_1 + l_3) \cdot cos(\theta_1) + l_2 \cdot cos(\theta_1 + \theta_2) \tag{3.6}$$

$$J_{22} = \frac{\partial z}{\partial \theta_2} = l_2 \cdot cos(\theta_1 + \theta_2) \tag{3.7}$$

Finally, the torques on the hip and knee joints are calculated as in equation 3.8:

$$T = J^\intercal \cdot F \tag{3.8}$$

In order to validate the Jacobian calculations, a simple PD controller was applied on the end-effector's position as follows:

$$T = J^\intercal \cdot (K_p \cdot \begin{bmatrix} x_d - x \\ z_d - z \end{bmatrix} + K_d \cdot \begin{bmatrix} -v_x \\ -v_z \end{bmatrix}) \tag{3.9}$$

Using (time derivatives from equations 3.1 and 3.2):

$$v_x = -(l_1 + l_3) \cdot sin(\theta_1) \cdot \dot{\theta_1} - l_2 \cdot sin(\theta_1 + \theta_2) \cdot (\dot{\theta_1} + \dot{\theta_2}) \tag{3.10}$$

$$v_z = \ (l_1 + l_3) \cdot cos(\theta_1) \cdot \dot{\theta_1} + l_2 \cdot cos(\theta_1 + \theta_2) \cdot (\dot{\theta_1} + \dot{\theta_2}) \tag{3.11}$$

To verify the correctness of the Jacobian, the leg is suspended in the air by the hip. An arbitrary x_d and z_d are chosen and appropriate gains ($K_p = 20$, $K_d = 5$ are applied, and the leg goes to the desired position, indicating that the Jacobian is correct.

4

Results

4.1. Single-leg simulation

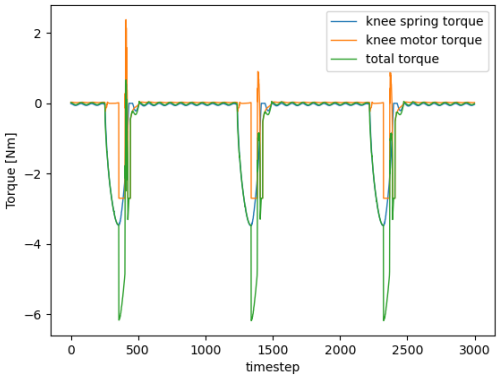
4.1.1. Mechanics

For each configuration, the KD gains and the spring constant were tuned experimentally, starting from the ones calculated in Table 2.1. The spring constants that gave the best results are shown in Table 4.1, along with the corresponding results. The tuning was done to reach the highest position possible.

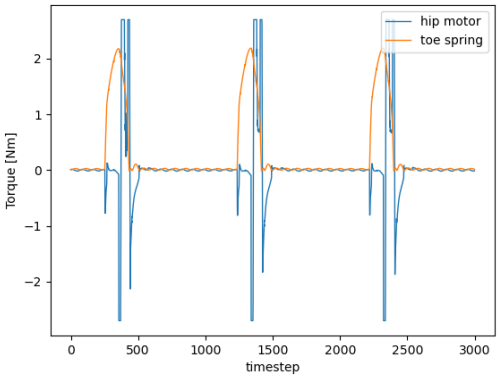
configuration	A	B	C
Spring constant k	1000	1900	400
Knee torque saved [%]	62.2	67.4	62.3
Total torque saved [%]	44.4	59.8	44.1
Height reached [m]	1.25	1.37	1.13

Table 4.1: Results for the three spring configurations

The torque profiles at the knee, hip and toe for 3 consecutive jumpgs using configuration A are shown in Fig. 4.1. The three control phases are very distinguishable. First there is a loading phase where the motors are off and the spring stores energy. Then, as all elastic energy is stored, the motors turn on to push down. It appears that they immediately saturate, meaning the maximum torque is applied and it would not be physically possible to push harder with the actuators used. From the rapid changes in the hip torques, it can be deducted that the hip motor does most of the work for keeping the toe below the hip. Finally, when the leg takes off from the ground, the knee and hip motors have a very short spike in order to reposition the foot below the hip and be ready for landing, then they deactivate. Overall the torque applied at the toe by the spring is consistent with the one applied at the knee, which is to be expected. During flight, the spring system wiggles a little bit because of the small toe positioning spring, which ensures that it is in a good position for landing. This wiggle is very small in comparison and does not cost energy to the motors.



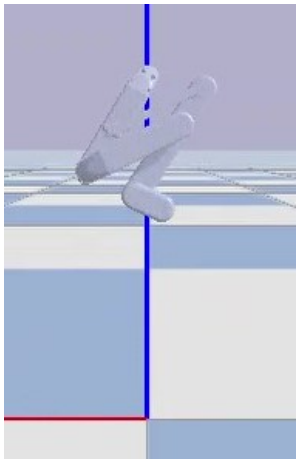
(a) Torque applied at the knee by the spring and the motor, with configuration A



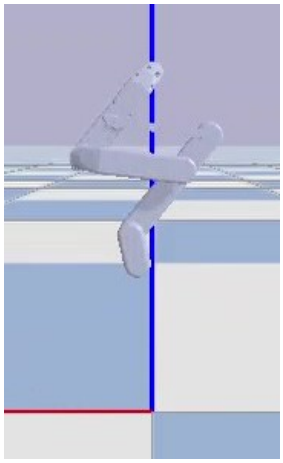
(b) Torques applied at the hip by the motor, and at the toe by the spring

Figure 4.1: All torques present in the system during three consecutive jumps

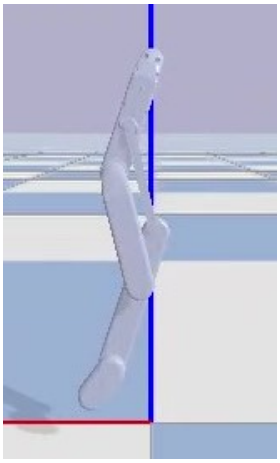
In order to make sure the toe position spring works as expected, the leg was suspended in the air and moved manually. As shown on Fig. 4.2, when the leg is retracted, the toe moves back as expected to keep the spring slack. Then when it is extended again, the toe moves back to the rest position for landing.



(a) flexed



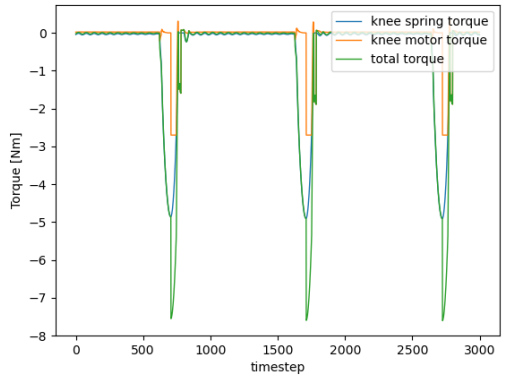
(b) half-flexed



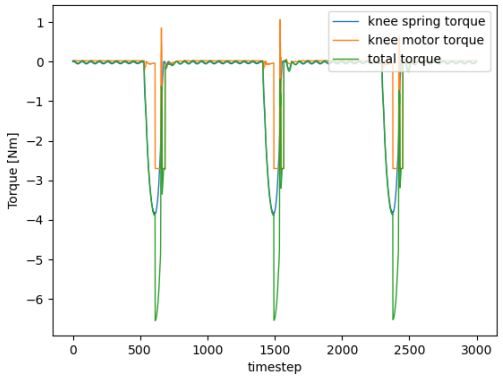
(c) extended

Figure 4.2: Toe retraction for different leg positions

The torque profiles for the knee joint of configurations B and C are shown in Fig. 4.3. They are extremely similar to the ones of configuration A, with the only difference being the height of the spring torque curves, which only depends on the height of the jump.



(a) Torque applied at the knee by the spring and the motor, with configuration A



(b) Torque applied at the knee by the spring and the motor, with configuration A

Figure 4.3: Motor and spring knee torques for configurations B and C

4.1.2. Maximum height

As shown in Table 4.1, the three configurations were able to jump higher than the leg without springs, which jumped 1.10m high. Configuration C was only slightly better, whereas A and B jump much higher. So even before considering the energy savings, adding springs allows to jump higher.

It appears that the optimal gains for the three configurations are very close to the ones calculated in Table 2.1. There is actually a very big correlation between the jumping height and the required K. Of course, a higher stiffness can store more energy and allow bigger jumps. And for the leg to jump higher, K needs to be bigger to be able to slow and stop the leg as it lands, otherwise it will slam to the ground. Therefore a higher K does not really make the leg jump higher, it simply **allows** it to jump higher and catch its weight during landing. However in order to jump high, the leg needs to start pushing at a low position so that the motors accelerate for longer and the spring has more energy to give out. The problem resides in the fact that in order to reach a low position, the legs has to first fall from a high point, otherwise the spring does not contract to its maximum potential and the next jump is not as high as it could be. This is a looping problem that comes from the control policy of not activating the motors during landing. If the motors were used to contract the leg during landing on top of using a higher K, more energy could be stored and a higher jumping height could be achieved, though it might be less energy efficient.

A second interesting point is that the two configurations using the toe mechanism, A and C, have a smaller maximum height than B. This suggests that the toe mechanism is less efficient compared to the same configuration with the tendon not attached to the toe. This is because some of the energy stored in the spring is used at the toe joint, which is less useful for jumping than at the knee joint. Of course that does not mean that applying torque at the toe is bad (animals do it all the time at push-off), but rather that the energy stored is more efficiently used at the knee than at the toe.

4.1.3. Energy saved

The proportion of energy saved is calculated by comparing the torque applied by the motors with the total torque applied by the motors and the springs. From Table 4.1, the proportion of energy saved at the knee joint is over 60% for all three configurations, which is very good. When taking into account the torque applied at the hip joint, where there is no spring, the total energy saved is between 44 and 60% depending on the configuration. This shows that parallel elastic actuation can be a very valuable addon to robotic legs.

Configuration B saves more energy than the others, for two reasons. The first one is that it jumps higher. For higher jumps, the leg contracts more, therefore more energy is stored and provided by the spring. As the motor torques saturate immediately when jumping, increasing the spring torques directly increases the proportion of energy provided by the spring.

The second reason is related to the proportion of energy saved when taking into account the hip torques. For configurations A and C, torque is applied at the toe, which has a tendency of pushing it forward instead of staying in place. To avoid that, the torque at the toe joint can be reduced by changing

the pulley ratios. Because the foot is pushed forward at pushoff, the controller needs to bring it back below the hip during flight, which is mainly done by rotating around the hip. Therefore configurations A and C increase the hip torque needed because the toe torque displaces the leg forward. Still, even in that case, over 44% of the total energy is saved.

Some care should be taken when choosing the pulley ratios, especially the ratio between r_1 and r_2, the toe and knee pulley. When $\frac{r_1}{r_2}$ is decreased, the torque at the toe is reduced compared to the torque at the knee, as shown in equations 2.1 and 2.9. As mentioned before, it might be good to reduce the amount of torque at the toe, so this ratio should be made smaller. However, if it is too small, it can reduce the freedom of movement in swing phase, as shown in Fig. 4.4. When retracting the leg, the toe is pulled back in order to avoid having to go against the spring. When the ratio $\frac{r_1}{r_2}$ is reduced, the toe is pulled back more for a given retraction. Therefore reducing the ratio too much could limit the range of motion in which the leg can be moved without resistance in swing phase. This is more of a problem in configuration C than A, because the spring displacement on the range of motion is around twice as big because there are two knee pulleys instead of one, so a flexion creates a bigger displacement in the tendon. In this project, complete freedom is wanted, and it is achieved with the chosen pulley radii on configuration A.

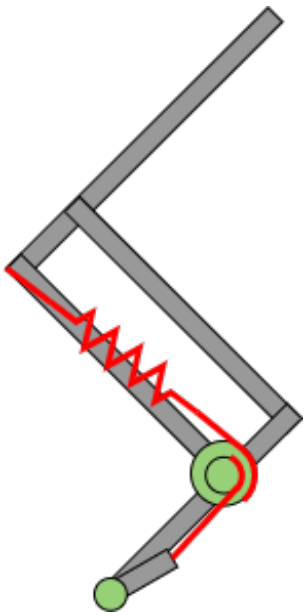


Figure 4.4: Toe completely pulled back, thus engaging the spring mechanism

Overall configuration B yields the best results of the three. However it does not incorporate the toe mechanism at all, therefore the resistance of the spring during swing phase would lead to a big energy loss, especially with the high stiffness of the spring. In this experiment it does not matter much because the leg stays straight during flight, but in a bounding forward scenario, there would be losses. As this project is focused on studying the incorporation of the toe mechanism, further analyses are done using configuration A, as its energy saving is similar to C but the height reached is significantly higher.

4.1.4. Sensing delay resilience

In real robotics, sensing and actuation is not instantaneous. Therefore it is interesting to investigate how much delay the system can stand. As explored by Shafiee et.al [2], a big part of the delay comes from sensing. The sensors can have a different update frequency than the controller, and their signals often need to be processed, creating delay. This was incorporated to the simulation as a delay on the desired force at the end effector. A range of delays was tested until the leg was unstable. The maximum delay supported by each configuration is shown in Table 4.2.

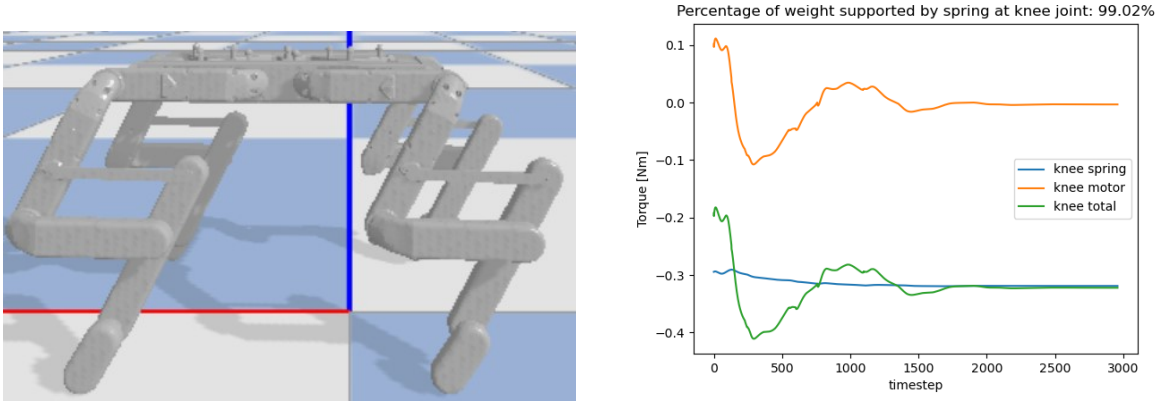
configuration	A	B	C
max delay	17ms	19ms	18ms

Table 4.2: Results for the three spring configurations

The leg with spring and tendon can withstand up to 19ms of delay when jumping. Above that, it becomes unstable and the toe tends to shoot forward or backward because the controller is late for controlling its position. As the control loop has a frequency of 1000Hz, it means that the controller can accommodate receiving data up to control loops late. Interestingly, the three configurations were able to jump a few centimeters higher with a very small delay (2-3ms), so this could be incorporated in the control policy when trying to jump high.

4.2. Quadruped simulation

In order to verify that the spring configuration can support the weight of a robot in a good manner, four identical legs were mounted in simulation on the main body of the Solo robot. The simulated robot is shown on Fig. 4.5a as it is in PyBullet. The control loop implemented is a simple PD controller on each leg to make the robot stand at a desired height. Each leg is equipped with configuration A of springs. By choosing the appropriate spring constant, almost the entire weight of the robot is supported by the springs. For a desired height of 35cm and a spring constant of 150, the torques applied by the motor and spring at the knee for one on the legs is shown in Fig. 4.5b. As the robot is only standing in place and not jumping, the spring constant is much lower because it does not need to absorb jumps from several body heights.



(a) Full robot model in simulation

(b) Torques applied at the knee by the motor and spring

Figure 4.5: Full body simulation results

The system easily moves to the desired position. It starts with all legs at a 32cm length with the toe 9.3 centimeters in front of the hip (this is the base position of the leg with all joints at 90°). As can be seen on Fig. 4.5b, the motor applies some torque at the beginning to move the leg to the desired position (24cm length, toe below the hip), then no torque is needed any more as the spring supports the weight. At rest, more than 99% of the weight is taken by the spring, which means almost no energy is needed to keep the robot in that position. All energy can be used for moving the robot around.

The robot tends to stand on its toes. When jumping, because of the inertia and the fact that a high tension is needed to slow the robot down, the toe is pressed to the ground. However when simply standing, much lower torques are needed, therefore even with a smaller spring constant, the torque at the toe is still too high and the robot stands on its toes. It should be noted that in a simply standing scenario, the toe does not bring anything to the system so it would not be useful on a real robot that only stands and can be removed.

A simple jumping test was performed with the full robot too, with springs of stiffness of 600 for more jumping power. The same controller is used as with the single leg, i.e. the spring stores energy until the robot is at its lowest point, then the legs push down to jump, as can be seen on Fig. 4.6. As it is

a common controller for all legs, meaning each leg pushes with the same force, the orientation of the robot is not really controlled. Only the elasticity brought by the springs softens the landing, so smaller jumps are attempted. Thanks to the springs, the robot manages to hop stably up to 59cm, over twice its resting height, and the springs are responsible for 60.7% of the energy necessary at the knee joint. Though the results are encouraging, the toe torque was still a little too high and the robot half-stands on its toes.

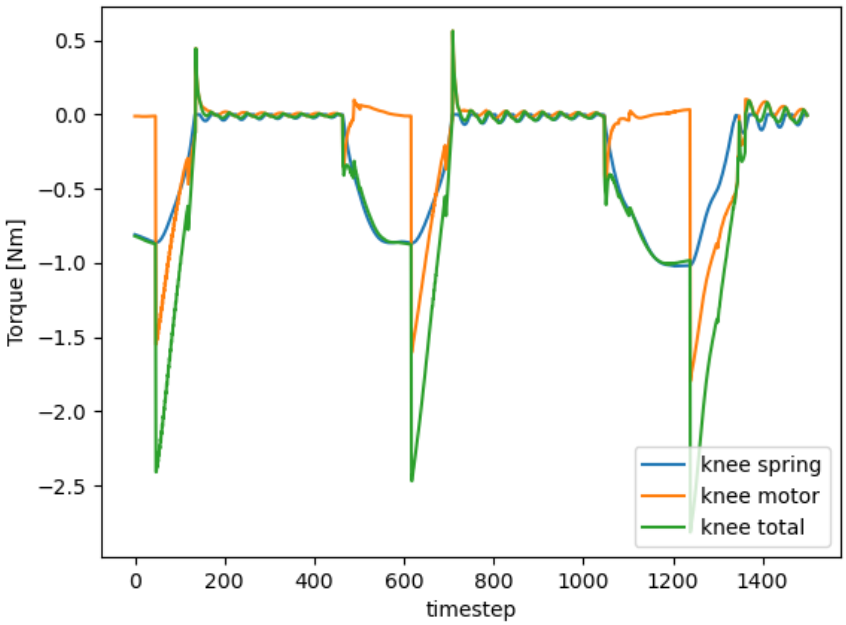


Figure 4.6: Torques applied at the front left knee by the motor and spring

4.3. Final design

From the results, it was decided that between configurations A and C, which are the two that include the toe clutch mechanism, configuration A was the most promising. It achieves very similar energy savings for the jumping task, but manages to jump significantly higher. A refined mechanical design was done, with the parameters shown in Table 4.3, as can be seen on Fig. 4.7.

A ratio of 2 was used for $\frac{r_1}{r_2}$, as it gave the best results in simulation. The pulley radii are 1.8cm for the toe and 3.6cm for the ankle. These were chosen to be consistent with the link sizes of the Solo robot and to be able to use mostly the same ordered parts. As a result, the spring for a drop of 3 body heights should have a stiffness of 1243N/m, as calculated from equations 2.1 and 2.14.

parameter	spring constant	toe pulley r_1	knee pulley r_2 = r_3
value	1243N/m	18mm	36mm

Table 4.3: Dimensions for the three final, real design

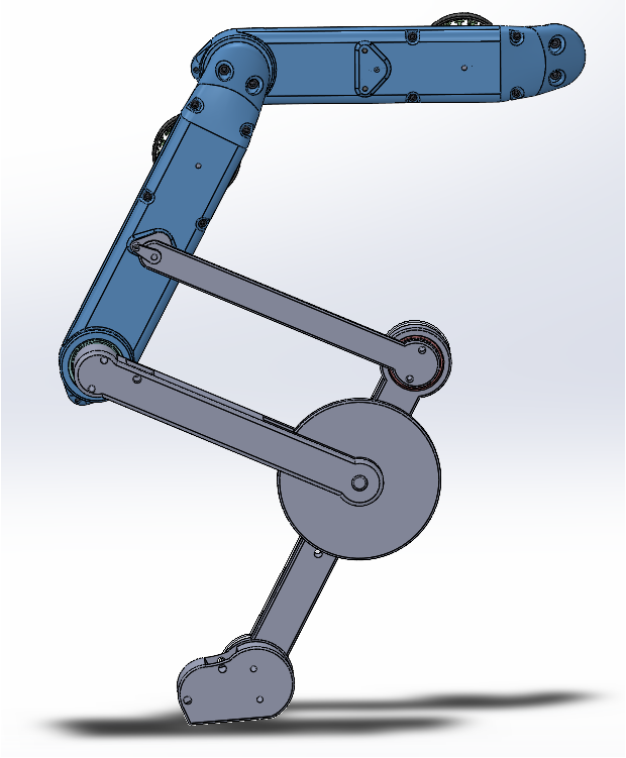


Figure 4.7: CAD model ready for building, with attachments for springs and tendons

5

Discussion

5.1. Discussion on the results

Overall the results are very encouraging. They show that a very large large part of energy can be saved when jumping by incorporating parallel elastic actuation. Even with a relatively short actuation window, the leg is able to jump to very respectable heights, comparable to what felines can do.

It should be noted that with the gains used, which are relatively high, the leg is almost fully extended at rest, even with the weight of the robot pushing down. This is because they were calculated to catch the whole weight of the robot from a fall of 3 body heights. This is not a problem in a jumping scenario, as all energy is used for the jumps. But in a standing or walking scenario, where the average height of the robot is more constant and lower, the high springs would keep the leg extended as the weight of the robot is not sufficient to flex the leg. Therefore in such scenarios, smaller spring constants should be used, which would reduce the amount of energy saved for jumping. In general, the springs used need to be appropriately chosen depending on the planned behavior of the robot, in particular the average desired body height.

In industry, most uses of robotic quadrupeds are for inspection and other tasks in which the robot moves relatively slowly and is stable. For such tasks, adding small springs that would only support the weight of the robot would save a big part of the energy. It would probably not be useful to incorporate a toe clutch mechanism as the flight phase is very short and a lot of leg retraction is usually not necessary. However, for more agile tasks involving running, bounding or jumping, adding stiff springs can greatly increase the energy saved and potentially the maximum height. This should only be done for robots that do not need to walk at all or only very little, because stiff springs would force the robot in a very high and upright position.

In agile scenarios, the toe clutch mechanism would be valuable as the leg positioning movements in flight phase are much bigger than in walking. However in walking, it is better to remove that part and attach the tendon directly to the leg, as it is not useful and wastes energy as well as plops the robot on its toes.

5.2. Future work

The next task to do now is to actually build a leg, test it and compare the results to the simulations. Several spring stiffnesses should be compared for different jumping heights. Also, the toe produces some forward torque, which can be beneficial for forward gaits like bounding or running, so the exact impact of this should be looked into. The necessary files for building a robot can be found on the github page of this project, accessible here. They are ready for 3D-printing and the parts list is included. Additionally, the two actuator modules form the Solo robot should be built according to the instructions present on its website [5].

After testing, the first challenge to tackle is to modify the mechanism in order to reduce the torque applied at the toe. According to the simulations, it is too large and interferes with the movement when the task is less agile (like simply standing). This can be done by changing the pulley radii and introducing new elements to the mechanism.

It would also be interesting to investigate how much energy is saved on configurations with a smaller spring constant. Implementing such a configuration on a current industrial inspection robot, with a spring only supporting the robot's weight, can lead to big energy savings.

The second thing to do is to investigate the quadruped's movements for agile gaits. In particular, bounding and running seem to be two gaits where spring configurations and toe clutch can have a big impact, because these are high-energy, high-impact gaits where the leg needs to move a lot during swing. The energy saved by adding parallel elastic actuation should be quite high, and it will be interesting to see exactly how the legs behave.

It would be interesting and necessary to implement different, more complicated controllers. The simple controller used for testing in this project is purely reactive, and has no planning. By using more advanced controllers, it is possible to have more control on the leg and use its elasticity more efficiently. As the torques generated by the system are completely defined by the joint positions, and these positions are known anyway for control, it is possible to calculate virtual torques and take them into account in the controller and adapt accordingly.

Implementing such controllers has the possibility to enable more agile maneuvers for quadrupeds. Some precise jumps or bounds are currently not achievable by quadrupeds due to a lack of power, so harnessing the energy stored in springs can be a solution.

6

Conclusion

The proposed leg design with parallel elastic actuation demonstrates that adding bio-inspired elastic structures can greatly reduce the amount of energy necessary for jumping or standing. A 3-segmented pantograph model with a toe was simulated with three different spring-tendon configurations and a toe clutch mechanism. After evaluation, they all yielded important energy savings and increased jumping height. The simulations allowed to put into light the requirements for dimensioning the elastic and rotating elements, in particular the importance of tailoring the spring stiffness to the desired behavior (walking, running, jumping, ...). The toe clutch is a practical solution to dis-engage the mechanism during flight and facilitate actuation, though it should only be used on robots that mostly run, bound or jump repeatedly. Although the work was done in simulation, a complete mechanical model of a leg with elastic elements was designed and is ready to be built for testing. Overall this subject deserves more study and testing as the applications are numerous and can lead to breakthroughs in agile and energetically efficient robotics

All files related to simulation and mechanical design are available on this github repository.

Aknowledgements

I would like to express my very great appreciation to my supervisors Milad Shafiee and Guillaume Bellegarda for their valuable suggestions during the development of this project, as well as Erfan Etesami for this fruitful collaboration.

References

[1] *ANYmal robot, ANYbotics*. https://www.anybotics.com/anymal-autonomous-legged-robot/. Accessed: 2022-06-10.

[2] Milad Shafiee Ashtiani, Alborz Aghamaleki Sarvestani, and Alexander Badri-Spröwitz. “Hybrid Parallel Compliance Allows Robots to Operate With Sensorimotor Delays and Low Control Frequencies”. In: *Frontiers in Robotics and AI* 8 (2021). ISSN: 2296-9144. DOI: 10.3389/frobt.2021.645748. URL: https://www.frontiersin.org/article/10.3389/frobt.2021.645748.

[3] Alexander Badri-Spröwitz et al. “BirdBot achieves energy-efficient gait with minimal control using avian-inspired leg clutching”. In: *Science Robotics* 7.64 (2022), eabg4055. DOI: 10.1126/scirobotics.abg4055. eprint: https://www.science.org/doi/pdf/10.1126/scirobotics.abg4055. URL: https://www.science.org/doi/abs/10.1126/scirobotics.abg4055.

[4] Ryuki Sato et al. “Vertical Jumping by a Legged Robot With Upper and Lower Leg Bi-Articular Muscle–Tendon Complexes”. In: *IEEE Robotics and Automation Letters* 6.4 (2021), pp. 7572–7579. DOI: 10.1109/LRA.2021.3099226.

[5] *Solo robot, open-source project*. https://solo.pal-robotics.com/solo. Accessed: 2022-06-10.

[6] *SPOT robot, boston dynamics*. https://www.bostondynamics.com/products/spot. Accessed: 2022-06-10.

[7] Lei Wang et al. “Design and Implementation of Jumping Robot with Multi-Springs Based on the Coupling of Polyarticular”. In: *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. 2018, pp. 287–292. DOI: 10.1109/ROBIO.2018.8664808.

[8] Patrick M. Wensing et al. “Proprioceptive Actuator Design in the MIT Cheetah: Impact Mitigation and High-Bandwidth Physical Interaction for Dynamic Legged Robots”. In: *IEEE Transactions on Robotics* 33.3 (2017), pp. 509–522. DOI: 10.1109/TRO.2016.2640183.