

Semester Project at the Biologically Inspired Robotics group,
Swiss Federal Institute of Technology Lausanne
Spring semester 2009

Locomotion exploiting body dynamics on the Cheetah robot

Ivan Kviatkevitch

Assistant : Alexander Sprowitz

Responsible Professor : Auke Jan Ijspeert

May 28, 2009



**BIOLOGICALLY INSPIRED
ROBOTICS GROUP (BIRG)**

Abstract

This project aims at designing a simple model of the Cheetah robot under Webots [1] simulation software developed by Cyberbotics Ltd. Cheetah is a small, light weight quadruped robot that features three segmented legs, thus very pronounced body dynamics. The model should make abstraction of the complex physics of the legs but should still produce a close to reality behavior. Furthermore, a pre-developed Central Pattern Generator (further referred as CPG) will be implemented in order to actuate the robot. The CPG is capable of four different gaits (bound, pace, walk and trot). The interest of this work lies in the exploration of the various parameters that affect locomotion, like stance phase duration or the amplitude of oscillation for the legs. The goal is to determine the most efficient parameter setups for different gaits.

Contents

1	Introduction	2
1.1	Task description	2
2	Hardware characterization	5
2.1	The Cheetah robot	5
2.2	Servo characterization	7
2.2.1	Software	7
2.2.2	Experiments	7
2.2.3	RX-28	7
2.2.4	AX-12+	7
2.3	Comparison between official data and experimental data . . .	9
2.3.1	RX-28	9
2.3.2	AX-12+	10
2.4	Conclusion	10
3	Webots model	11
3.1	Overview	11
3.2	Morphology and mass distribution	14
3.3	Model limitations	14
3.4	Divergence with the real hardware	16
4	Controllers	16
4.1	Central Pattern Generator	16
4.2	Controlling the robot	17
4.2.1	Single retraction model	17
4.2.2	Double retraction model	18
4.2.3	Retraction length	19
4.3	Three segmented leg behavior	21
5	Experiments using the Webots model of the Cheetah robot	23
5.1	Single retraction	25
5.1.1	Bound gait	25
5.1.2	Pace gait	27
5.1.3	Walk gait	29
5.1.4	Trot gait	31
5.2	Double retraction	33
5.2.1	Bound gait	33
5.2.2	Pace gait	35
5.2.3	Walk gait	37

5.2.4	Trot gait	39
5.3	Conclusion	40
6	Conclusion and Future Work	40
7	References	42

1 Introduction

This report summarizes the results of a semester project performed during the spring semester of 2009. This project is a continuation of several previous projects done at BIRG (Simon Rutishauser [2] and Martin Riess [3]). During his project, Simon Rutishauser constructed the first version of Cheetah, which was extensively used during this project. Martin Riess created a close to reality simulation of the Cheetah robot; unfortunately, the model was too complex for the simulator, which resulted in long simulation times. We were able to reuse some of the elements of his model to produce the current model of the Cheetah robot under Webots.

In order to build a simulation model, we first need to characterize several hardware components present on the real Cheetah robot. In our case, the servos characteristics had to be determined in an experimental way and used to calibrate the simulation to the real robot. The next step was to build a Webots model. A pre-existing model created by Martin Riess during his project [3] served as a starting point. By replacing complex three segmented legs by simpler components (see the Webots Model chapter for more details) and using the controller to calculate the compression of the leg, we obtained a good approximation of the real hardware. A pre-existing CPG was implemented on the robot to produce several gaits. Systematic searches were performed on some parameters to determine the optimal setups for each gait.

1.1 Task description

The task consisted in three major parts :

1. Hardware characterization: The goal was to experimentally determine servo characteristics used on the Cheetah robot, special interest was to determine the maximum frequencies these servo could achieve.
2. Creating a Webots model of the Cheetah robot: The requirements here were to avoid using complex physics and components in the simulation, but to still have a close to reality behavior of the legs. In order to

achieve these goals, the leg length calculations had to be implemented inside the controller. the touch sensors on the feet were necessary in order to use the sensor feedback feature of the CPG and calculate the leg length.

3. Implementing a CPG: A CPG developed by L.Rhigetti in [4] had to be implemented on the robot. It had to be tested for different gaits and parameter settings. Systematic searches had to be performed to find optimal setups. Additionally, single and double retraction models were explored (more on this in the CPG chapter).

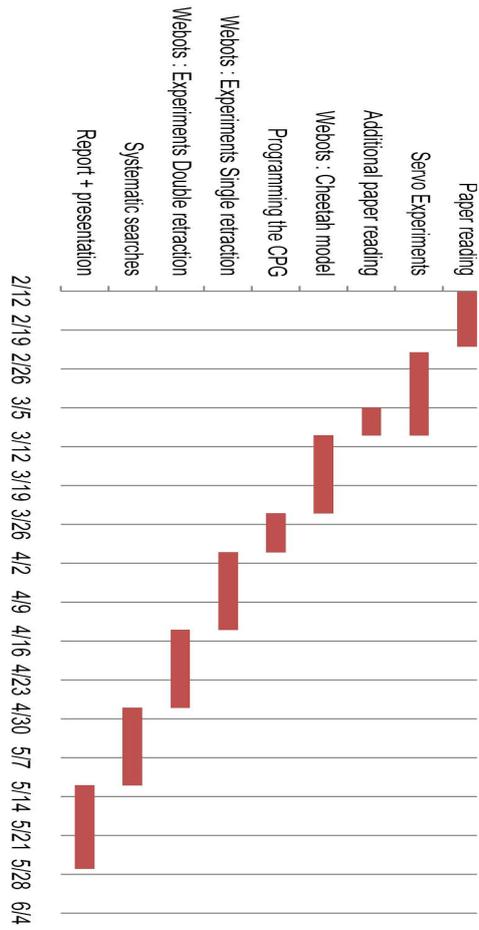


Figure 1: *Timetable for the project*

2 Hardware characterization

2.1 The Cheetah robot

The Cheetah robot is a lightweight quadruped robot with three-segmented legs; it is presented in fig.2. The first version of Cheetah was built by Simon Rutishauser during his semester project [2]. The robot has been upgraded several times since then, but the main features remain unchanged. Its main features are two actuated degrees of freedom per leg (knee and hip) and a passive compliant knee joints. Its leg relative dimensions are designed based on biological data available on small mammals, like cats and dogs. As observed in small mammals, the leg has only two degrees of freedom due to a pantograph mechanism used to keep the upper segment of the leg parallel to the lowest segment at all time. The passive compliant knee joints influence both the retraction and the extension of the leg by acting as a counter force during retraction and allowing extension of the leg without any actuation. More details can be found in the report on the construction of the Cheetah [2].

The robot is equipped with 8 servos. 4 servos are the Dynamixel AX-12+; these servos are used to actuate the hip joint of the robot. The rest of the servos are Dynamixel RX-28, and are used to actuate the knee joint via a Bowden cable assembly as presented on the hardware sketch in fig.3

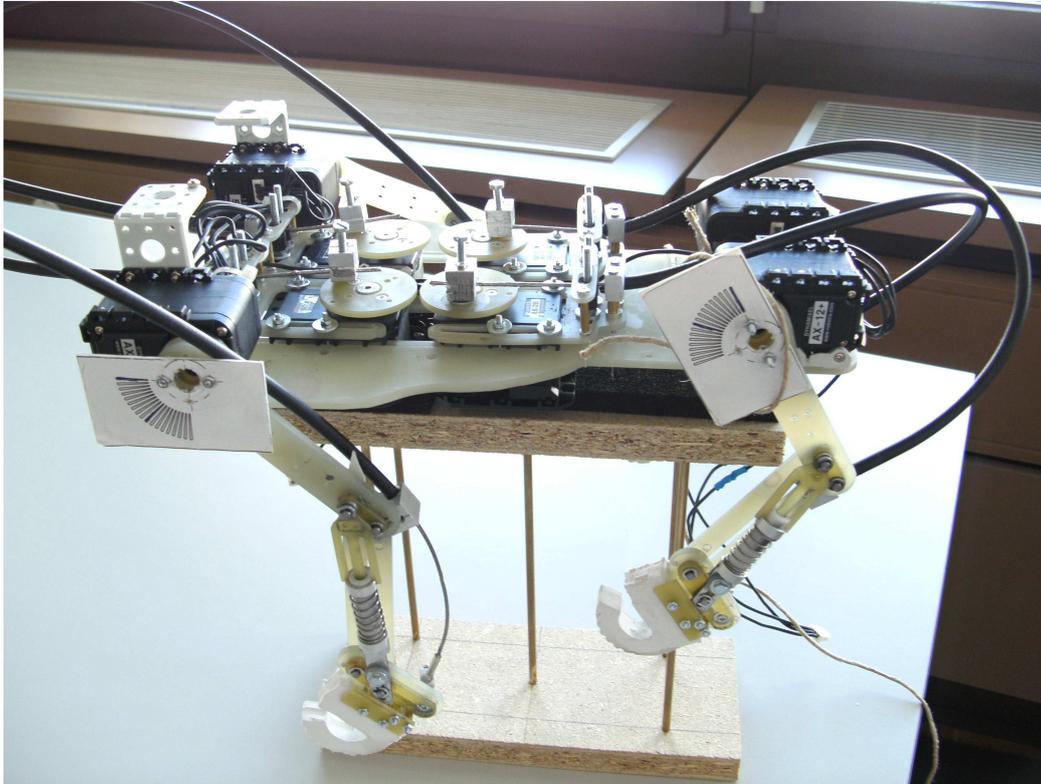


Figure 2: *Cheetah robot*

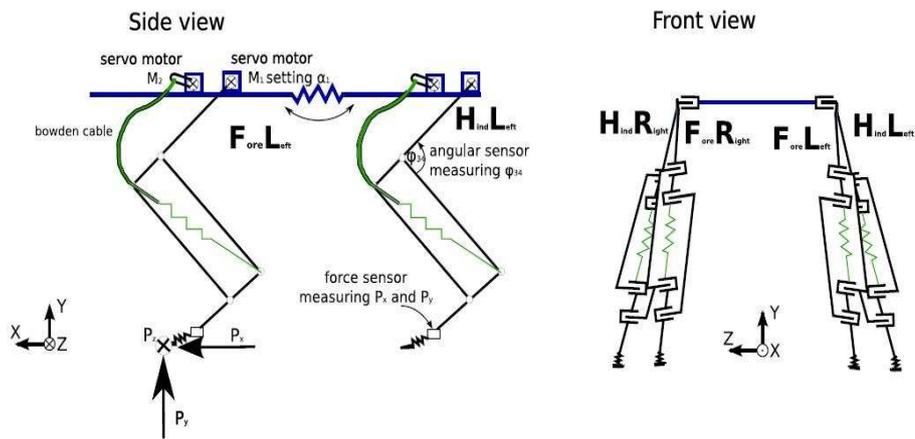


Figure 3: *Hardware sketch for Cheetah 1.0, taken from Simon's Rutishauser's report*

2.2 Servo characterization

2.2.1 Software

During the experiments the Bioloid Behavior Control programm (found at <http://www.robotis.com>) was used to command the servos. This program gives the ability to control rotation speed and position of the servos in position mode and rotation speed in continuous rotation mode.

Note that the software used with this experiments works as follows for the position mode: we give a desired position to the servo and wait for the moving bit to be set to 0. The moving bit in the servo is 1 whenever the servo is moving and 0 once it has stopped. Once the bit is 0, we input a new position.

2.2.2 Experiments

For the RX-28 servos, the relevant characteristic is the round per minutes (further referred to as RPM), as these servos are used in continued rotation mode. The impact of the voltage and the load factor on the RPM was determined.

For the AX-12+ servos, we explored the impact of the load and rotation speed on the frequency under constant voltage.

2.2.3 RX-28

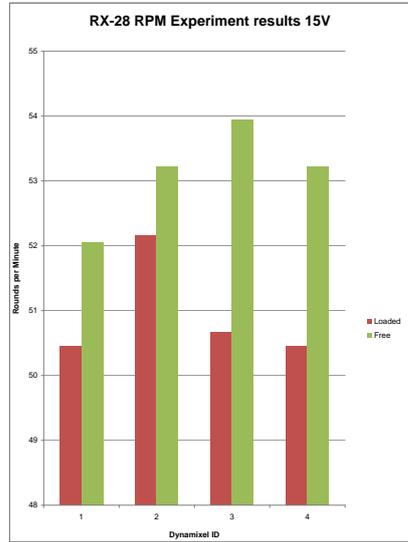
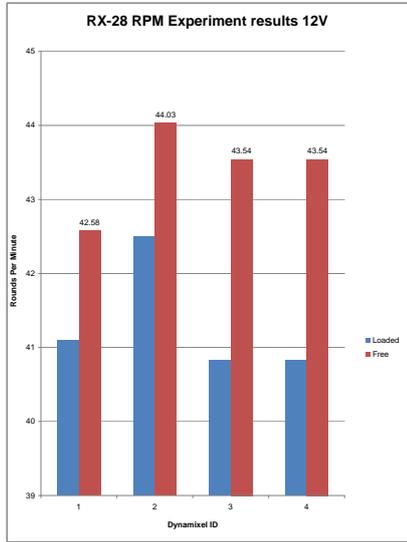
As stated above, we have determined the RPM for the RX-28 servos depending on voltage and load. We used the CM-2+ controller board for this experiment, which gave us the possibility to try out different voltages. The voltages that were used are : 12V, 15V and 17V. The load factors were : free (Bowden cables disconnected) and standard load (Bowden cables connected to the servos). The results are presented in Fig.4.

Remark : Dynamixel with id 2 was poorly connected properly to the leg, which explains it's abnormally high RPM with load.

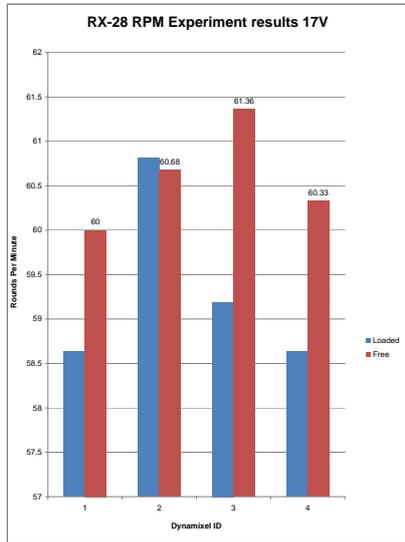
Precision : We admit ± 0.16 on rounds and ± 0.25 on time values.

2.2.4 AX-12+

For these servos we explored achievable frequencies under variable load and amplitude. The voltage was kept at 12V all along the experiment. The CM5 controller was used. We chose the following amplitudes : 40, 45 and 50 (in degrees). These values were chosen because the real robot's hip oscillation will have similar amplitude. Different speed control values were tested to

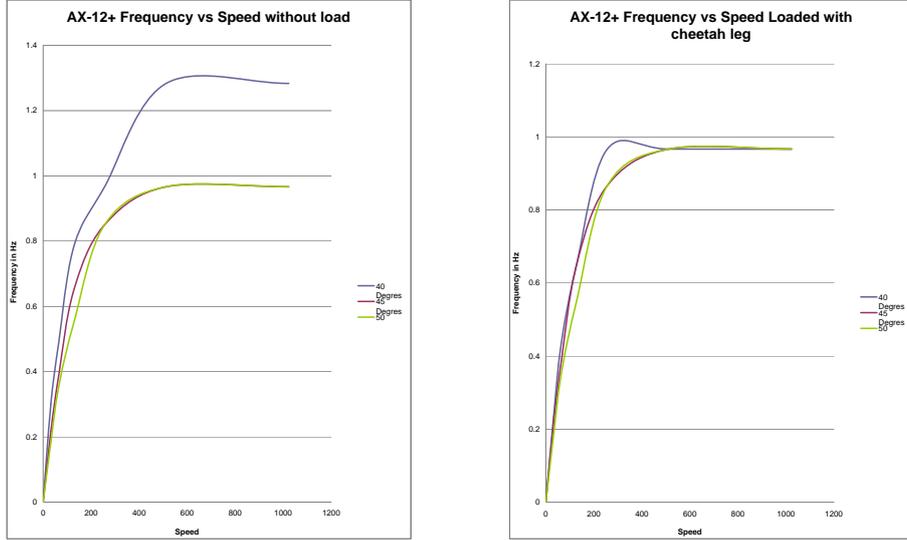


(a) *RX-28 RPM experiment under 12V* (b) *RX-28 RPM experiment under 12V*



(c) *RX-28 RPM experiment under 12V*

Figure 4: RX-28 Servo RPM for various loads and voltages



(a) *AX-12 frequency experiment under 12V unloaded* (b) *AX-12 frequency experiment under 12V loaded*

Figure 5: AX-12+ Frequency values for various amplitudes speed and load

establish their impact on the frequency and standard load was applied (cheetah’s leg) in one case, and the servo ran unloaded in the second case. The results are presented in Fig.5.

2.3 Comparison between official data and experimental data

2.3.1 RX-28

The official data for the RX-28 servo claims that at 12V the servo has a speed of 0.167 seconds for 60 degrees and at 16V the servo has a speed of 0.126 seconds for 60 degrees. Lets compare it to the experimental data we have collected. To convert the experimental data to a comparable value with official data we proceed as follows :

$$t_{round} = 6 \times t_{60deg} \quad (1)$$

where t_{rounds} is the time in seconds for 360 degrees and t_{60deg} is the time in seconds for 60 degrees.

$$t_{round} = \frac{t_{exp}}{rounds} \quad (2)$$

where t_{round} is the time in seconds for 360 degrees and t_{exp} is the length of the experiment.

By applying eq.1 and the 12V official data we get 1.002 seconds per round. By applying eq.2 to the results found during experiments we obtain 1.36 seconds per round at best (no load). Loaded the time obtained is 1.411 at best. The claimed value can be achieved at 17V with load.

For 16V, by applying eq.1 we get 0.756 seconds per round. From the table results we get 0.97 seconds per rounds at 17V with load. The claimed value was never achieved, even if we consider the imprecisions, we achieve 0.925 seconds per round at best with no load.

2.3.2 AX-12+

The official data states that AX-12+ servo has a speed of 0.196 seconds per 60 degrees at 10V. We were running this servo at 12V. To compare these values we need to apply eq.3 to the experimental data :

$$t_{60deg} = \frac{60}{2 \times f \times \alpha} \quad (3)$$

Where f is the frequency and α is the amplitude.

For 40 degree oscillations we get 0.58 seconds unloaded at maximum speed, and 0.77 seconds loaded at maximum speed. For 45 degrees we get 0.69 seconds both loaded and unloaded conditions at maximum speed. For 50 degrees oscillations we get 0.62 seconds for both loaded and unloaded conditions at maximum speed.

As we can see there is no way that these servos could achieve the claimed speed. It may be due to the fact that the speed announced by the manufacturer were calculated in the continuous rotation mode and we are using position mode on these servos.

Remark: the servos stop for a consistent amount of time at the end of an oscillation, which halves the performance expected from the official data. This maybe due to the fact that the software waits for the stop bit to be set to 0. Because we have no access to source code, we cannot determine precisely how and when this checking is done. By using some other software, there maybe a way to decrease the time the servo spends without moving.

2.4 Conclusion

We were using the RX-28 servos in continuous rotation mode during this experiments. The results suggest that these servos are too slow compared to the hip servos: we expect the swing phase duration to be around 0.62 seconds

with the current hip servos, but the speed of a round of RX-28 (a retraction cycle) is 1.411, which is unacceptably slow. Further on, we will assume that this servos are used in position control mode. We will now calculate the radius of the discs needed to perform a retraction of 17mm on the Bowden cable with 60 degrees oscillations.

By solving the triangle ABC seen on fig.6 we find a relation between r , l , the angle a and the distance the Bowden cable was retracted given in eq.4:

$$l + x = \sqrt{(r \sin(a))^2 + (r - r \cos(a) + l)^2} \quad (4)$$

Where l is the minimum distance between the fix point of the Bowden cable on the servo and its fixation on the body, r is the radius of the support attached to the back servo, x is the retraction of the Bowden cable and a is the angular position of the servo.

By solving this equation for r with $a = \frac{\pi}{3}$, $x = 0.017$ and $l = 0.02$ we obtain the minimum radius needed in order to be able to perform a full retraction with an amplitude of oscillation of 60 degrees. We obtain 2.85 cm, which seems reasonable and can be implemented on the actual hardware.

3 Webots model

3.1 Overview

The model is presented in fig.7. The starting point for this model was Martin Riess's model presented in [3]. The body geometry was preserved as much as possible with regard to the real hardware. The legs on the original model attempted to simulate the leg behavior on the model level, which resulted in use of complex components and a complementary physics plug in. The main disadvantages were the slow simulation speeds and a complex model. The advantages are a close to reality leg design which successfully simulates the non-linear behavior of the legs.

For this model we chose a different approach: we decided to make abstraction of the complex physics and geometry of the legs at the model level, and compute the compression values in the controller using sensory feedback from the feet of the robot. This approach yields a better modularity (the leg design can be changed at will with no changes to the model) and greatly decreases the complexity of the model, which allows us to run the simulations in real-time. The model introduces some abstractions of the real hardware which will be mentioned at the end of this chapter.

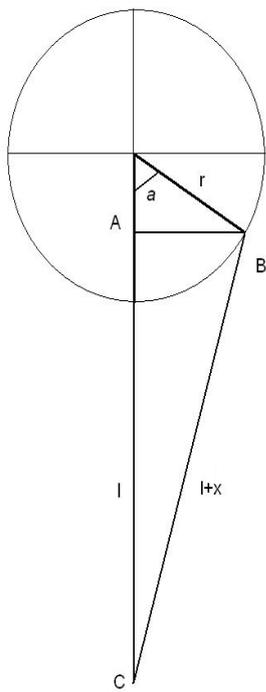


Figure 6: *Sketch of the back servo retracting the Bowden cable*

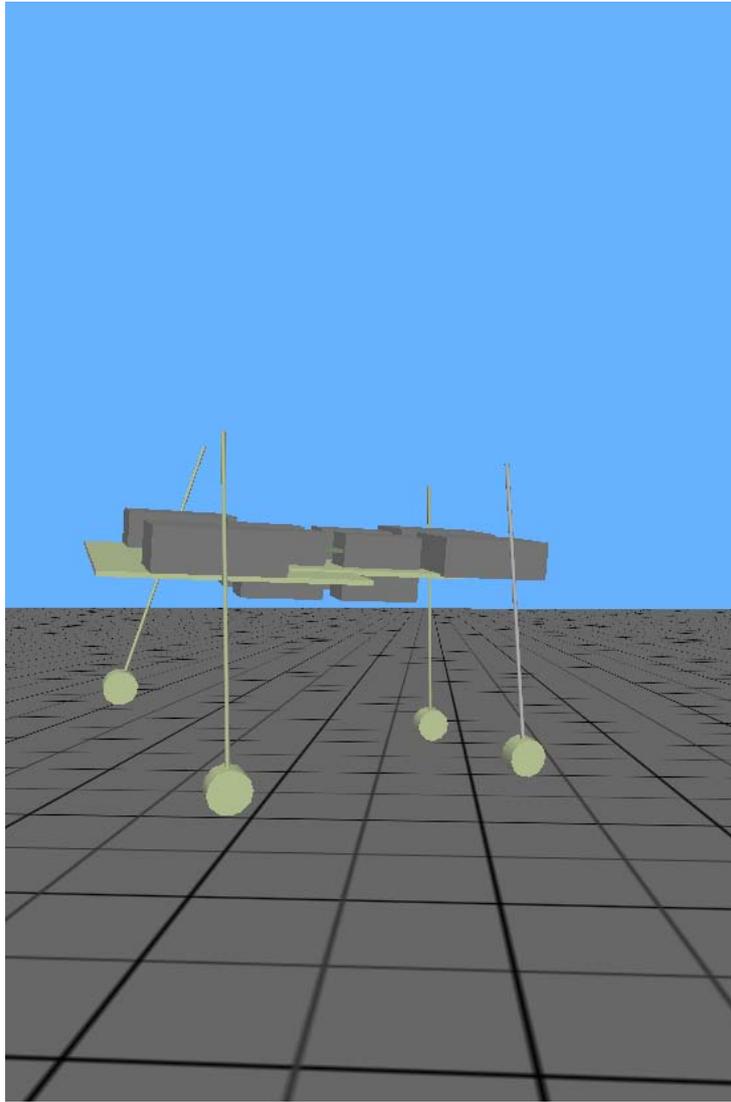
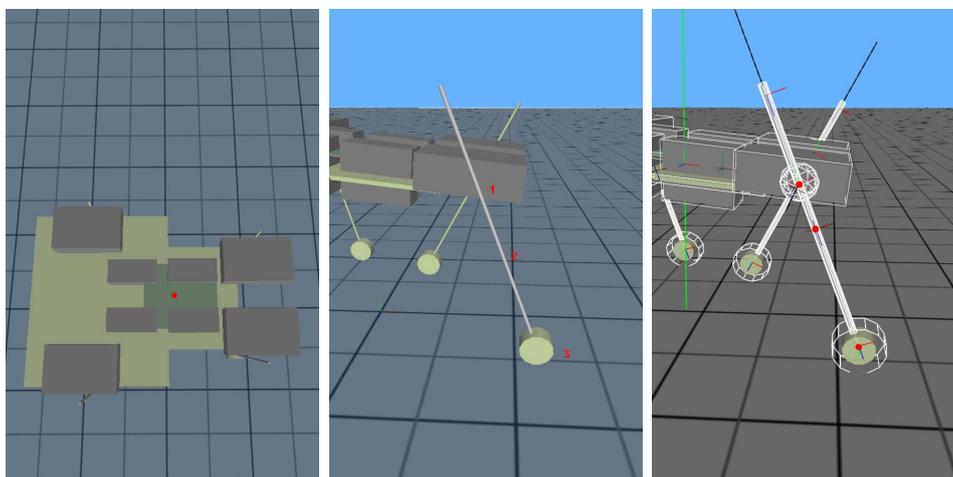


Figure 7: *Current model of the Cheetah robot under Webots simulation software*

3.2 Morphology and mass distribution

In this section we will present the model in a more detailed view and mention the mass distribution. As can see on fig.8(a) , the body of the robot is built using several box bounding objects. To further simplify the model, the body is considered by Webots as a single object with a mass of 700 grams. The center of mass for the body was chosen in such a way that the robot is able to stand on 3 feet while not actuated. As can see on fig.8(b), the leg is composed of three parts: a rotational servo (1), a linear servo (2) and a touch sensor (3). The mass of each component was fixed to 20 grams, to achieve a total of 60 grams per leg. The centers of mass are presented in fig.8(c).



(a) Cheetah model's back, red spot marks the center of mass
(b) Cheetah model's leg
(c) Cheetah model's leg with mass distribution marked in red

Figure 8: Various views of Cheetah model with centers of mass marked in red

3.3 Model limitations

The main limitation of the current model is the poor quality of sensor signal as can be seen on fig.9 We were expecting to see a smoother curve of pressure applied on the touch sensor. This is a major problem because this values are used in servo position calculations. If the values are not precise enough, we cannot expect the model to simulate correctly the real hardware behavior because the values are used in the leg position calculations. As will be presented later, an error of 1 Newton on sensor reading yields a difference of 2mm in leg length. If raw sensor values are used directly, the results are

catastrophic: the leg oscillates very fast vertically, which causes the robot to become unstable and lose contact with the ground during stance phase.

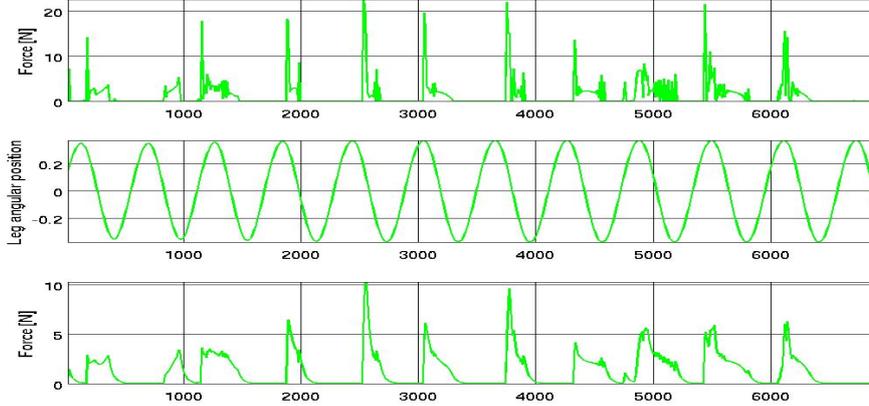


Figure 9: *Raw sensor values reported by Webots touch sensor, followed by a Hip trajectory, followed by an interpolated sensor values used in the calculations*

The solution for this problem is to slow down the linear servos of the model as much as possible. We assume a maximum retraction factor of 65% of the maximum leg length during the experiments. The maximum leg length is 150mm for this version of Cheetah, which means that the linear servo will have to produce a movement of 100mm to retract the leg and 100mm movement to extend it. By using eq.5 we can calculate the minimum acceleration required for the linear servos of the model to still be able to perform a retraction and an extension during the swing phase.

$$x = \frac{at^2}{2} \quad (5)$$

By fixing t to be 0.15 sec (half of the fastest swing phase the real servos can perform) we get an acceleration of 8.88 m/s^2 in order to be able to perform the 100mm position change in 0.15s. To be safe, we fix the acceleration parameters of the linear servos of the model to be 9 m/s^2 . This value would allow us to perform a full retraction cycle during 0.3 seconds.

An additional way of filtering the signal is to use eq.6 on the sensor input.

$$sensor_{smooth} = sensor_{smooth} + \frac{sensorvalue - sensor_{smooth}}{k} \quad (6)$$

$sensor_{smooth}$ is set to 0 in the beginning of the simulation and is altered every simulation step using the reported sensor value. The k parameter determines the speed of convergence between the sensor data and $sensor_{smooth}$,

the current model has k fixed to 5.0. This yields a result presented in the bottom graph of fig.9. These values were used in the controller calculations.

3.4 Divergence with the real hardware

We tried to produce a model that is as close as possible to the real hardware, but some components present of the real robot had to be simulated at controller level.

The most important divergence is the knee actuation. The real robot uses rotational servos to pull on the Bowden cable that in turn retracts the spring assembly. This setup introduces a non linearity between the servo position and the retracted distance. The model uses a single linear servo to simulate the knee joint actuation. This servo cannot simulate the non linear relation between it's position and the retracted distance as observed on the real model. Additionally, the real leg geometry is ignored in the model, so the real hardware may not be able to assume certain configurations the model could (for example leg retractions greater than 100mm).

4 Controllers

4.1 Central Pattern Generator

In order to actuate the robot, we will use a pre-existing CPG developed by L. Righetti in [4]. This CPG uses a modified Hopf oscillator. A servo i follows the differential equations presented in eq.7:

$$\begin{aligned} x'_i &= \alpha(\mu - r_i^2)x_i - \omega_i y_i \\ y'_i &= \beta(\mu - r_i^2)y_i - \omega x_i + \sum k_{ij}y_j \\ \omega_i &= \frac{\omega_{stance}}{e^{-by} + 1} + \frac{\omega_{swing}}{e^{by} + 1} \end{aligned} \tag{7}$$

where $r_i = \sqrt{x_i^2 + y_i^2}$, α and β are positive constants defining the time of convergence to the limit cycle, b is a positive constant that is chosen to be high and k_{ij} are coupling constants from oscillators j to i taken for the coupling matrix for the current gait. $\sqrt{\mu}$ defines the amplitude of oscillation of x_i and y_i . The frequency of oscillation is given by ω_i which switches between ω_{swing} and ω_{stance} .

The main feature of this CPG is the capability to produce multiple gaits (bound, pace, walk and trot). There is a possibility of using the sensory

feedback to enhance the stability of a gait. Unfortunately, because of the quality of sensor signal on our model, we were unable to use this feature effectively. It is also convenient to change the stance and swing duration independently.

4.2 Controlling the robot

We now need to transform the CPG output to some values the servos accept. For the hip servos the equation is pretty straight forwarded and is given in eq.8.

$$position_{hip} = y_{cpg} \times \alpha + offset \quad (8)$$

Where y_{cpg} is the y variable of the CPG, α is the maximum amplitude of oscillation of the hip. y_{cpg} oscillates with amplitude $\sqrt{\mu}$. There are two ways of controlling the amplitude of oscillation of the leg: changing α or changing μ . For our convenience during the experiments, we were manipulating α directly and leaving μ at 1.0.

The issue is more complex for the knee servos. We will examine the possible models to resolve this issue. Two models are present in the literature are presented in the following chapters.

4.2.1 Single retraction model

The first approach presented in [10] is the following: the leg should be fully extended during the whole stance phase; retracted at the end of the stance phase and kept this way through the whole swing phase. We call this approach single retraction.

In order to implement this approach on the robot, we use the function presented in eq.9 as mentionned in [9]

$$k(x, y) = \begin{array}{ll} 0 & \text{if } y \geq 0 \text{ (stance phase)} \\ 10(1 - \|x\|) & \text{if } y < 0 \text{ and } \|x\| > 0.9 \\ 1 & \text{else (swing phase)} \end{array} \quad (9)$$

The function $k(x, y)$ corresponds to the factor of the compression of the pantograph mechanism, it is 0 if the leg is fully extended as during stance phase and 1 if the leg is completely compressed, as during most of the swing phase. x and y are CPG outputs. The relation between the spring length and the leg length is not linear but is of little interest in the moment because the leg is either fully contracted or extended most of the time.

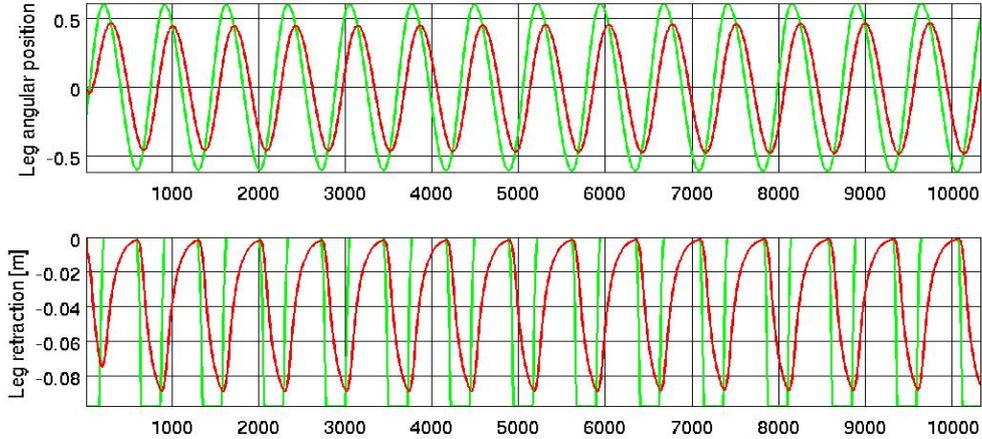


Figure 10: *In green, the trajectory generated by the CPG for the single retraction model. In red, the actual servo trajectory without passive elements (robot suspended in the air)*

4.2.2 Double retraction model

Recent research by Andre Seyfarth and Hugh Herr ([5] and [6]) show that during running mammals tend to retract their leg at touchdown. Based on this research, we will be using a second model that we call double retraction. The figure fig.11 presents the desired trajectory for the knee motors.

In order to implement the double retraction model on the robot, we will be using the function given in eq.10 for the stance phase ($y_{cpg} > 0$). l_λ corresponds to the relative length of the leg, 0 being a full contraction and 1 being full extension and x being the CPG output. Assuming the *hipmaxangle* variable to be fixed to 35 degrees, the maximum retraction during stance phase is achieved when $x = 0$ with $l_\lambda = 0.745$.

$$l_\lambda = \frac{\cos(\text{hipmaxangle})}{\cos(x \times \text{hipmaxangle})} 0.91 \quad (10)$$

$$k(x; y) = 1 - l_\lambda$$

The function $k(x; y)$ is similar to the one used on the single retraction model, it corresponds the factor of compression of the leg. $k(x; y)$ is 0 for $x = 1$ and $x = -1$ which means that the leg fully extended at touchdown and at takeoff. $k(x; y)$ has a maximum at $x = 0$, the leg is contracted by 25.5% of its maximum capability at that time.

For the swing phase ($y_{cpg} < 0$) we will be using the equations given in eq.11

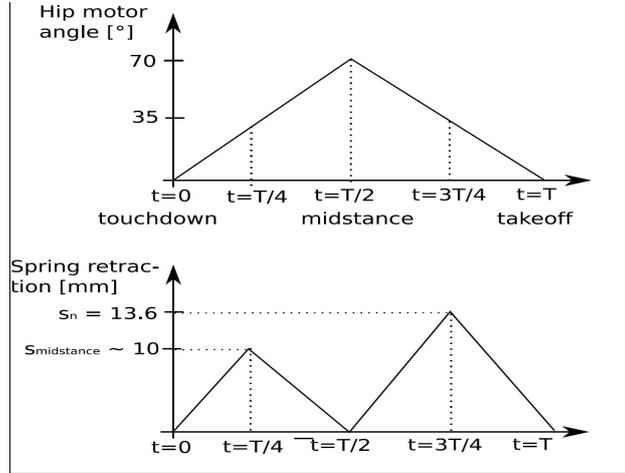


Figure 11: *Desired trajectory for the knee servos using the double retraction model, taken from S.Rutishauser's report p.38*

$$k(x, y) = \begin{cases} 1 & \text{if } x \leq 0.9 \text{ and } x \geq -0.9 \\ 10(1 - x) & \text{if } x > 0.9 \\ 10(1 + x) & \text{if } x < -0.9 \end{cases} \quad (11)$$

This equations look similar to the ones used in the single retraction model, which is no surprise because the double retraction model must perform the full retraction during the swing phase just like single retraction model. Figure fig.12 presents the trajectory generated by the CPG using the double retraction model and the actual motor trajectory without passive elements (no contact with the ground).

It is interesting to note that the double retraction pattern was also observed during gaits using the single retraction model. As we can observe on the fig.13, at touchdown the passive elements of the leg cause it to retract, which is exactly what we wanted to achieve using the double retraction model.

4.2.3 Retraction length

Webots servo position system is illustrated in fig.14. Note that the linear servo position in Webots is simply the negative of the desired retraction distance.

Now that we have the compression factors for different models we can

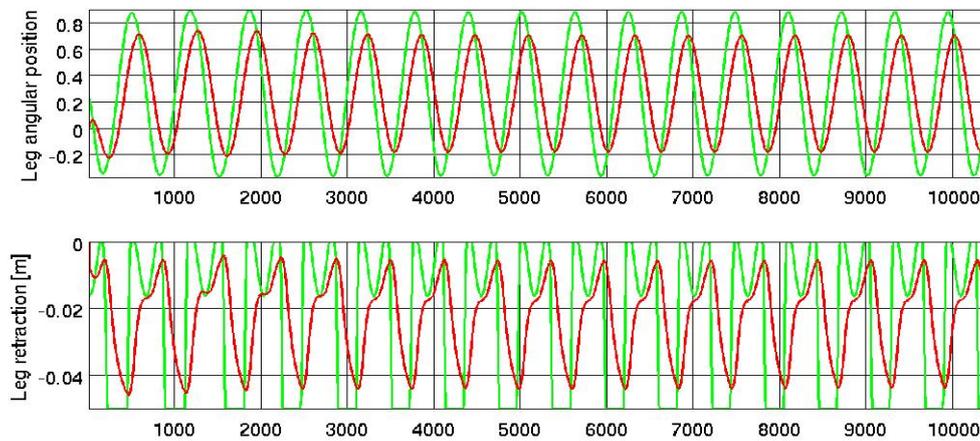


Figure 12: *In green, the trajectory generated by the CPG for the double retraction model. In red, the actual servo trajectory without passive elements (robot suspended in the air)*

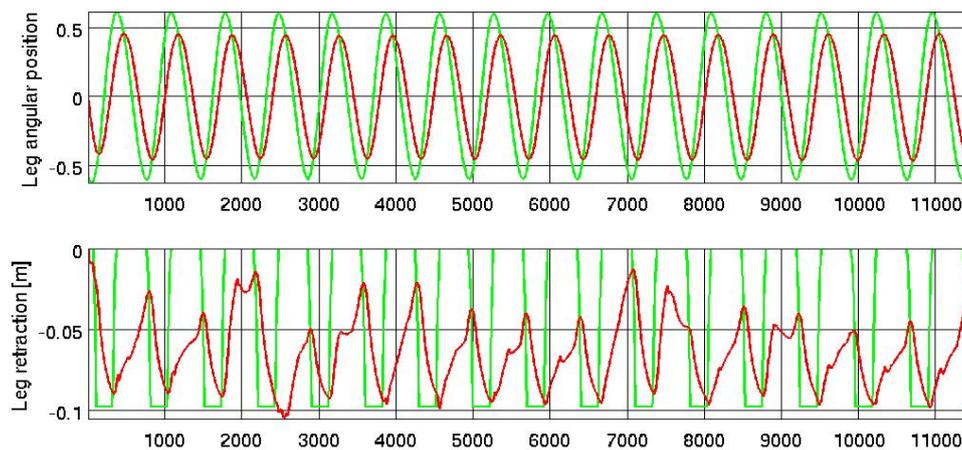
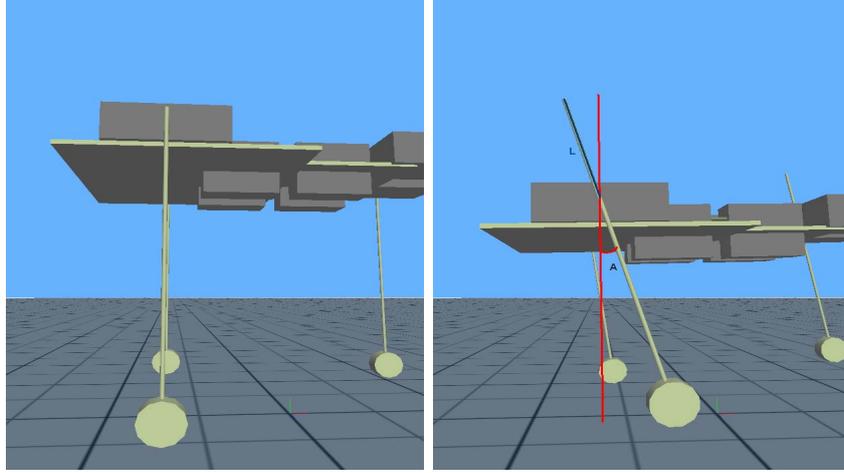


Figure 13: *In green, the trajectory generated by the CPG for the single retraction model. In red, the actual servo trajectory with passive elements.*



(a) Cheetah's leg at rest, linear and rotational servos are at 0 position
 (b) Cheetah's leg moved, linear servo position is $-L$ meters, rotational servo position is A degrees

Figure 14: Webots servo positions

establish the equations that transform the CPG output to the desired retraction length of the leg. The retraction length is given in eq.12

$$retraction_{knee} = -k(x; y) \times l \times l_{\lambda min} \quad (12)$$

Where $k(x; y)$ is the functions discussed above, l is the maximum leg length and $l_{\lambda min}$ is the maximum leg retraction factor.

4.3 Three segmented leg behavior

In order to implement the three segmented leg behavior on our model, we first need a function that would allow us to translate the force applied to the leg to the corresponding leg retraction. We will use the approximation of the experimental data presented in fig.15. We approximated this function by two linear equations given in eq.13

$$force = \begin{cases} 75.6 \times length - 1.05 & \text{if } length < 0.055 \\ 461.11 \times length - 22.26 & \text{else} \end{cases} \quad (13)$$

In order to calculate the final servo position we need to take into account two forces : the gravity force coming from the sensors, and the force applied by the servo on the leg at a given moment in time. By applying eq.14 we obtain the position we need to feed to the servos in the Webots model.

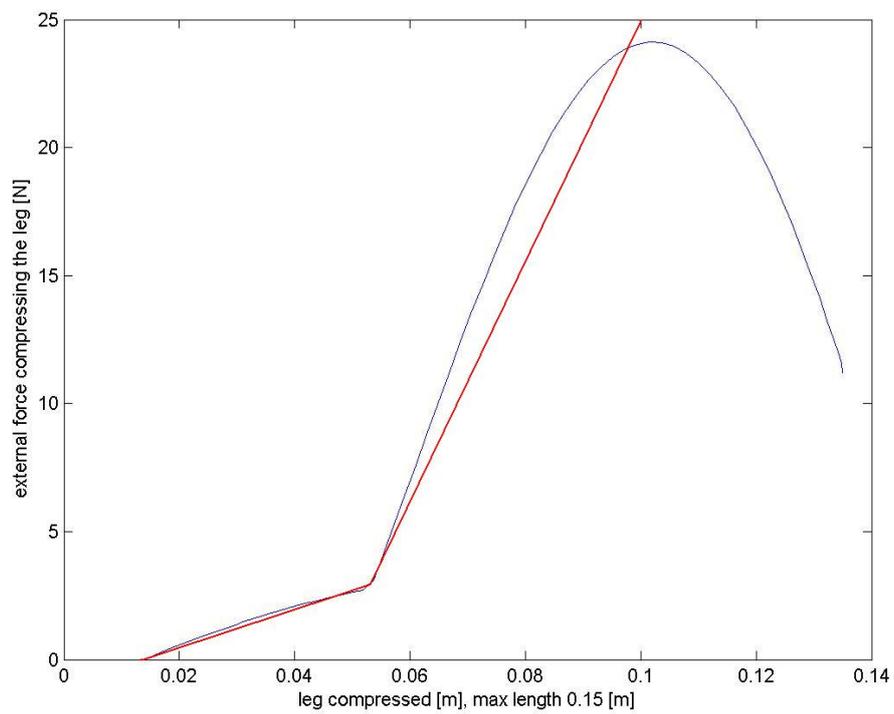


Figure 15: *In blue, solution for the static equilibrium equations for the three segmented leg presented in [2] p.27. In red, the approximation used in the controller*

$$position_{knee} = F^{-1}(F(d) + sensorforce) \quad (14)$$

Where the function F is the function presented in eq.13. d is the retraction length given in eq.12 and $sensorforce$ is the value read from the touch sensor.

5 Experiments using the Webots model of the Cheetah robot

Now that we described all the components present in our system, we can start exploring the locomotion possibilities of the model. As stated above, the CPG supports four gaits: bound, pace, walk and trot. Furthermore, we need to explore two retraction model we presented in the previous section. The parameters of interest for us are the stance duration, hind leg offset, maximum hip oscillation amplitude and maximum leg retraction factor (mentioned earlier as $l_{\lambda min}$).

The stance duration is an important parameter because we suspect it to have a great impact on the speed of locomotion [7]. It is important to note that the minimum stance phase duration achievable by the current servos is 0.2 in theory, and 0.3 in practice, the searches for a stance phase faster than that are purely hypothetical. Hind leg offset is presented in fig.16, and proved to have major impact on certain gaits. The maximum leg retraction factor is of interest to determine the necessary speeds for the back servos and to determine its impact on stability of locomotion. For the amplitude of oscillation of the hip, $\sqrt{\mu}$ was explored. To convert from the values reported on the searches to values in degrees, we need to apply eq.15.

$$degrees = reported_value \times 70 \quad (15)$$

As mentioned in [7], the swing phase duration is kept constant for different speeds of locomotion. We fixed this parameter to be 0.32 seconds for all of our experiments.

In this section we will describe the systematic searches performed on various pairs of parameters and try to find some optimal setups for each gait. The metric for the experiments will be the absolute distance on the x axis, we will thus penalize the gaits that are not making the robot go straight ahead. For each setup, three experiments were performed, and the mean was reported on the systematic search graphs. If the robot fell down at any point, a 0 was awarded. General tactics for fixing the two unstarched parameters

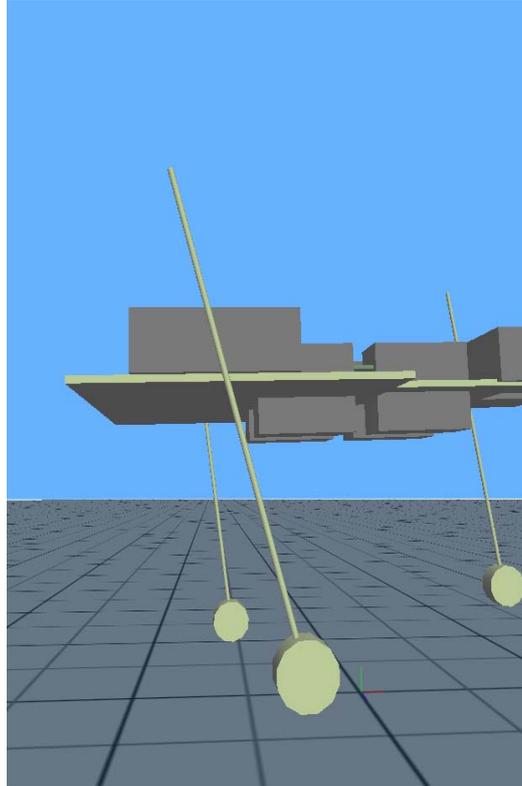


Figure 16: *A hind leg of the Cheetah robot using a non-zero offset, the rotational servo is at 0 position*

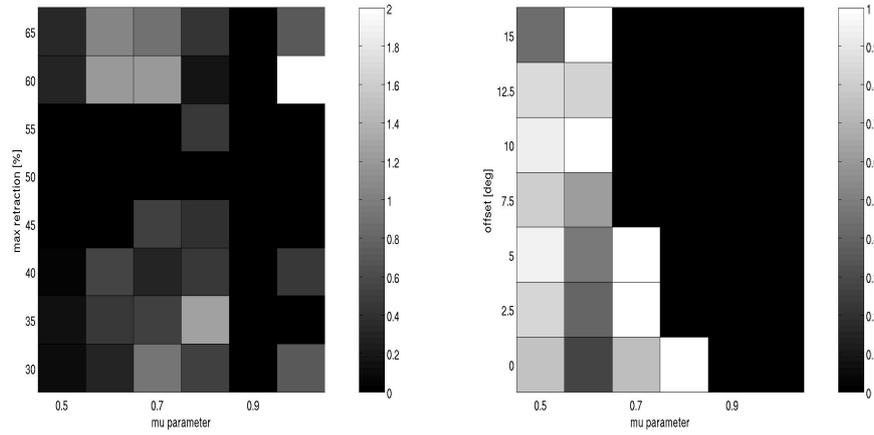
during a search was to perform low granularity searches before hand to get an intuitive idea about an acceptable value for the remaining parameters.

5.1 Single retraction

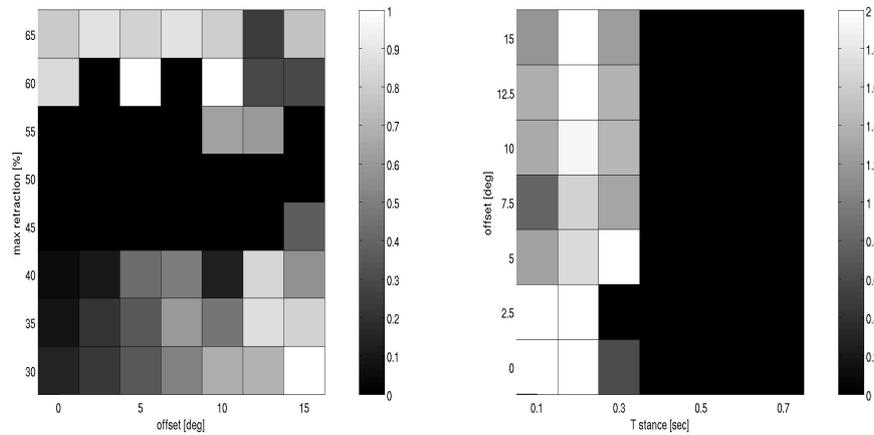
In this section we will present the results obtained using the single retraction model.

5.1.1 Bound gait

The systematic searches are presented in fig.17. As we can see on the fig.17(a), there are some good values for the couple $\sqrt{\mu}$ and max retraction factor. Good results were found for large max retraction factors (60% and higher). The amplitudes in range 0.6-0.8 yield acceptable results. Fig.17(b) tries to determine the values of amplitudes that can support an offset. As we can see, large amplitudes do not allow us to use any offset, because the robot becomes unstable and falls on its back. In fig.17(c) we investigate the impact of the max retraction factor and the offset using a relatively small amplitude on the speed. This search confirms our supposition that we need to use a large max retraction factor coupled with no offset, or a smaller retraction factor with some offset. Fig.17(d) illustrates the main problem that this gait presents: the servos need to be fast, not higher than 0.3 second stance phase. If the servos are slower, the robot spends too much time in an unstable state (all the weight on the hind limbs) and tends to fall on its back. The maximum speed achieved with this gait is $0.204m/s^2$. We can enhance the stability of the gait by exploring the smaller amplitudes, but the results for this amplitudes are consistently slower (around $0.12m/s$). Best parameter setup : 0.3 second stance duration, 60% max retraction factor, 5 degrees offset and 70 degrees amplitudes of oscillation.



(a) Amplitude versus max retraction factor search. Stance duration fixed to 0.3 seconds and offset fixed to 5 degrees
 (b) Amplitude versus offset search. Stance duration fixed to 0.3 seconds and max retraction factor fixed to 40%

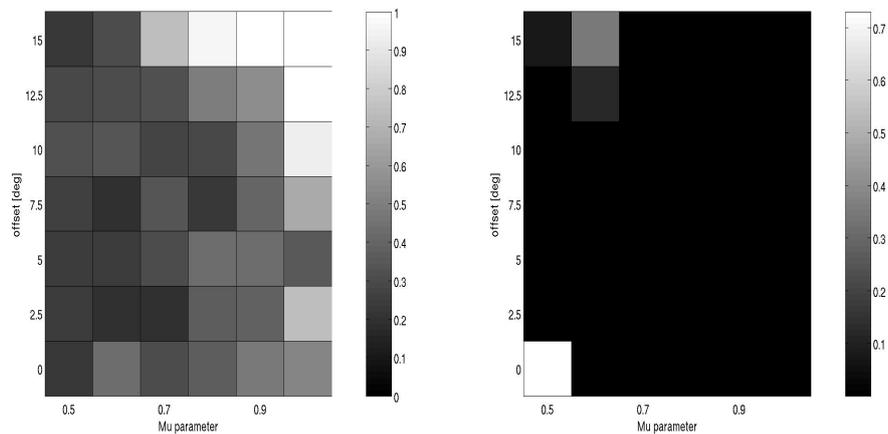


(c) Offset versus max retraction search. Stance duration fixed to 0.3 and $\sqrt{\mu}$ fixed to 0.6
 (d) Stance duration versus offset search. Max retraction factor fixed to 60% and $\sqrt{\mu}$ fixed to 1.0

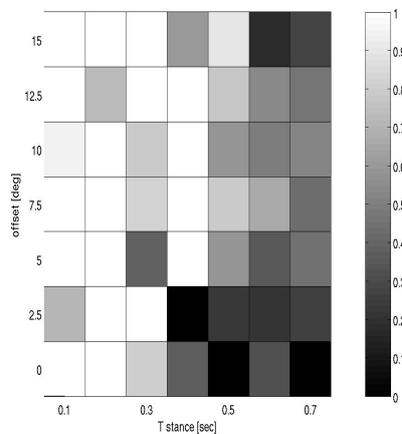
Figure 17: Systematic searches on various parameters for the bound gait

5.1.2 Pace gait

The systematic searches for this gait are presented in fig.18. We encountered a major problem with this gait: the spring seemed too stiff for this gait, the rolling movements were building up in the body and the robot fell on the side for almost all setups of amplitudes and offsets as presented in fig.18(b). This is maybe due to the fact that we filtered the sensor inputs, because no such problems were encountered on the real robot [8]. The solution we used to solve this issue, was to amplify the sensor input. Mechanically it is equivalent to reducing the spring stiffness. The sensor information was doubled for this experiment only. The results resemble much more the pattern that the real robot presents for this gait. As we can see on fig.18(a), there are a lot of good values for this gait, we observe better results for large amplitudes and large offset values. Fig.18(c) confirms the hypothesis that this gait should be used with an offset on the hind limbs. We observe a lot of good values, even for slower servos. The max retraction factor was not explored for this gait, because of the sliding behavior observed for lower values of the retraction factor. The best result obtained with this gait is $0.21m/s$. The gait is a bit slower than a bound gait in general, but is much more stable, almost any parameter setup will yield a naturally looking gait. Best parameter setup : 0.3 second stance phase, 65% max retraction factor, 12.5 degrees offset and 70 degrees oscillations.



(a) *Offset versus amplitude search.* (b) *Offset versus amplitude search. Stance duration fixed to 0.3 seconds and Stance duration fixed to 0.3 seconds and max retraction factor fixed to 65%, re- max retraction factor fixed to 65% reduced spring stiffness*

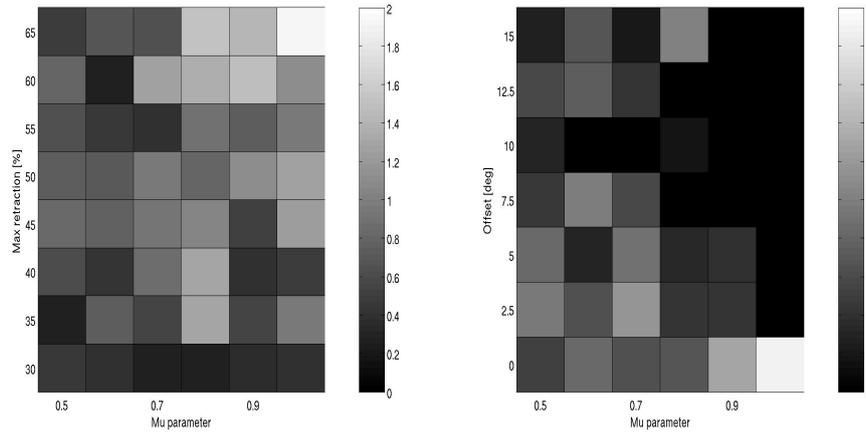


(c) *Stance duration versus offset search. max retraction factor fixed to 65% and $\sqrt{\mu}$ fixed to 1.0, reduced spring stiffness*

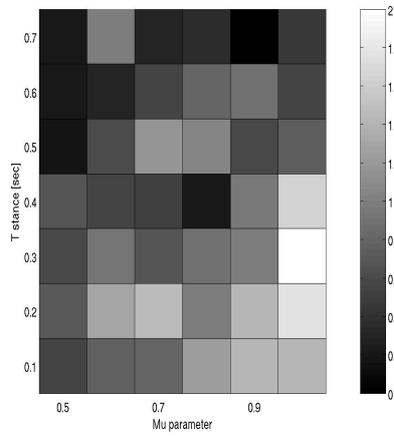
Figure 18: *Systematic searches on various parameters for the pace gait*

5.1.3 Walk gait

The searches for the walking gait are presented in fig.19. As we can see on fig.19(a) there is a clear maximum at 65% max retraction factor and 1.0 values for $\sqrt{\mu}$. On fig.19(b) we can see the same pattern, a clear good value at $\sqrt{\mu} = 1.0$ and offset = 0. We can also see that this gait is very unstable with a non-zero offset. The search presented in fig.19(c) confirms the need for a biggest amplitude possible: we observe good values for various speed only with the largest amplitude. Qualitatively the gait looks good, but by looking at the searches we can see that there is not a lot of good parameter setups available for this gait, but the few stable ones are performing very good. Maximum speed achieved is $0.228m/s$. The advantage of this gait is the absence of body movements during locomotion, the coupling of the feet compensate the forces generated during motion which yields better stability. The pattern displayed by this gait is comparable to what was achieved with the real robot as presented in [8] ($0.25m/s$ and a clear preference for large amplitudes). Best parameter setup : 0.3 second stance phase, 65% max retraction factor, 0 offset and 70 degrees oscillations.



(a) *Amplitude versus Max retraction factor search. Stance duration fixed to 0.3 seconds and offset fixed to 0* (b) *Offset versus amplitude search. Stance duration fixed to 0.3 seconds and max retraction factor fixed to 65%*

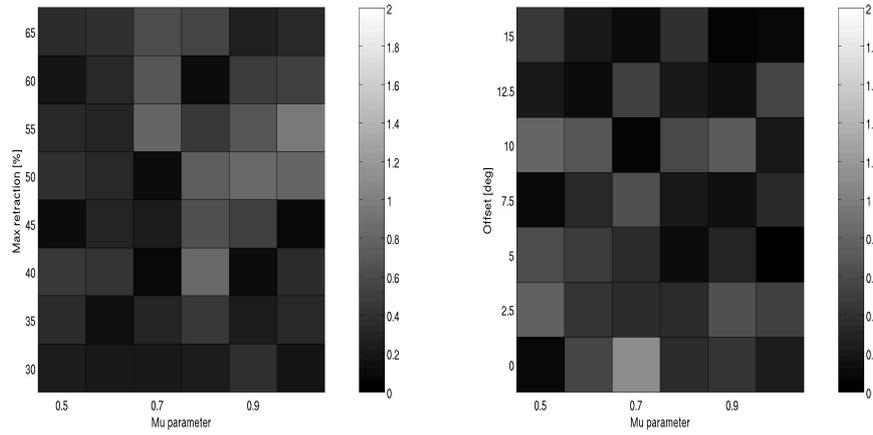


(c) *Stance duration versus amplitude search. max retraction factor fixed to 65% and offset fixed to 0*

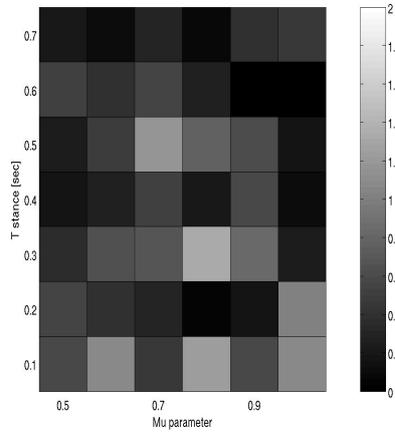
Figure 19: *Systematic searches on various parameters for the walk gait*

5.1.4 Trot gait

The systematic searches for this gait are presented in fig.20. We observe erratic patterns on all the searches, with no clear preferences for a particular setup. This is mainly due to the fact that this gait almost never goes straight, which is very penalizing with the metric we are using. We also observed convergence problems with this gait: the controller converges much faster to a pace or to a bound gait. The gait sometime converges to a strange unknown gait (front limbs seem to perform a pace gait and hind limbs are performing what seems to be a bound gait), which causes it to be highly unstable. The clear spots in the searches correspond sometimes to this poorly converged gaits and should not be considered as usable parameter setups. The rare times the controller successfully converged towards a trot gait, we had a stable gait (the robot was going in circles, but still in a stable way). If the convergence problems can be resolved, this gait could perform well on this robot. We may need to change the coupling matrix or tune the convergence parameter of the CPG and redo the searches for this gait if we want to have consistent data.



(a) Amplitude versus Max retraction factor search. Stance duration fixed to 0.3 seconds and offset fixed to 0
 (b) Offset versus amplitude search. Stance duration fixed to 0.3 seconds and max retraction factor fixed to 65%



(c) Stance duration versus amplitude search. max retraction factor fixed to 65% and offset fixed to 0

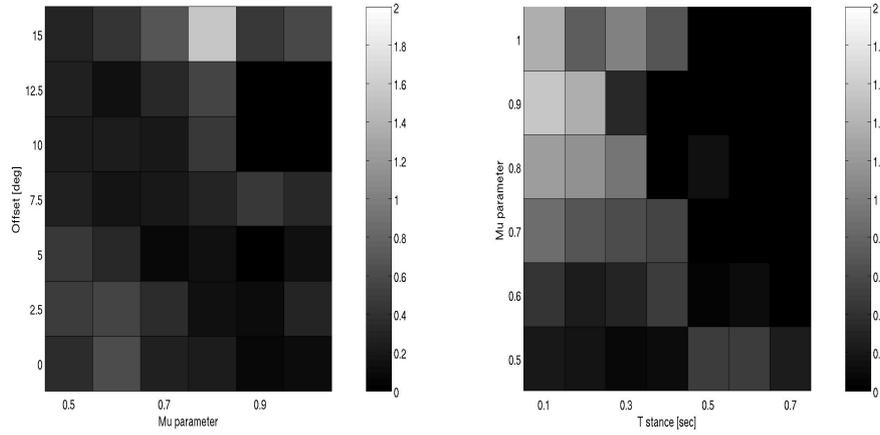
Figure 20: Systematic searches on various parameters for the trot gait

5.2 Double retraction

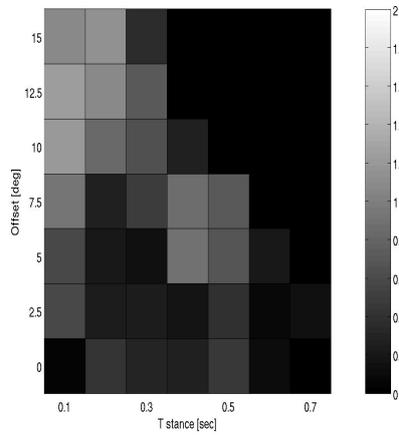
In this section we will present the results obtained using the double retraction model. For the double retraction experiments, the maximum retraction factor was fixed to 65% due to the fact that lower values would cause the feet to stay in contact with the ground during the swing phase.

5.2.1 Bound gait

The systematic searches for the bound gait using the double retraction model are presented in fig.21. As we can see on the fig.21(a), there is one clear maximum at 15 degrees offset and $\sqrt{\mu} = 0.8$, we also observe that the rest of parameters yield poor speeds but the robot doesn't fall down. By comparing this search to the one presented in fig.17(b), we can see that the single retraction gait is faster but is also more unstable: many offset values cause the robot to fall down, in contrast with the double retraction model that is generally slower but more stable. Fig.21(b) illustrates the preference for large amplitude values for this gait, we still have the problem present in the single retraction model: the servos need to be fast. The last search presented in fig.21(c), shows that it is possible to perform this gait with slower motors if the offset is tuned correctly, the robot achieved $0.126m/s$ with 0.5 second stance. In contrast with the single retraction model, which required fast servos, this model allows for a wider range of parameters at cost of speed. The best value achieved using this gait is $0.155m/s$. Best parameter setup : 0.2 second stance phase, 15 degrees offset, 65% max retraction factor and 56 degrees oscillations.



(a) *Amplitude versus Offset search.* (b) *Amplitude versus stance duration*
Stance duration fixed to 0.3 seconds and search. Offset fixed to 15 and max re-
max retraction factor fixed to 65% *traction factor fixed to 65%*

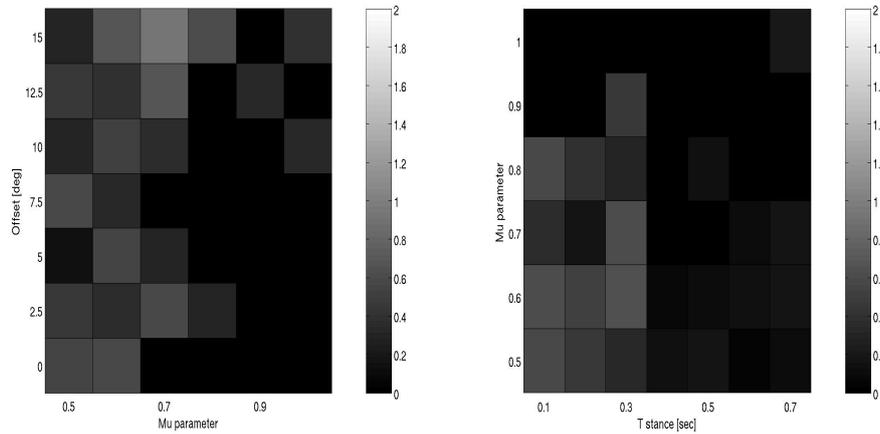


(c) *Stance duration versus offset search.*
max retraction factor fixed to 65% and
 $\sqrt{\mu}$ fixed to 0.8

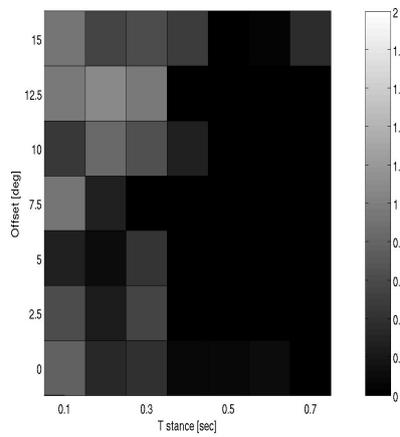
Figure 21: *Systematic searches on various parameters for the double retraction bound gait*

5.2.2 Pace gait

The searches for this gait are presented in fig.22. The single retraction version of this gait suffered from spring stiffness being too high. We were interested to see if the double retraction model would resolve this issue. As we can see on fig.22(a), the robot is not falling down any longer for all couples of amplitude and offset as it did with a single retraction model. This may be due to the fact that the body is closer to the ground during stance phase. The robot can't use large amplitudes with this setup; this is mainly due to the problem present in the single retraction model: horizontal body movements become too strong and the robot falls to the side. We suspect that reducing the spring stiffness, as we did in the single retraction model, can greatly enhance the results of this gait. The searches presented in fig.22(b) and fig.22(c) show us that the servos need to be fast (0.3 second stance phase) in order for the robot to remain stable. Once again, this issue has been encountered with the single retraction model, and was successfully resolved by reducing the spring stiffness. The best values achieved with this gait is $0.112m/s$. Best parameter setup : 0.3 second stance, 12.5 degrees offset, 65% max retraction factor and 49 degrees oscillations.



(a) *Amplitude versus Offset search.* (b) *Amplitude versus stance duration*
Stance duration fixed to 0.3 seconds and search. Offset fixed to 12.5 and max re-
max retraction factor fixed to 65% *traction factor fixed to 65%*

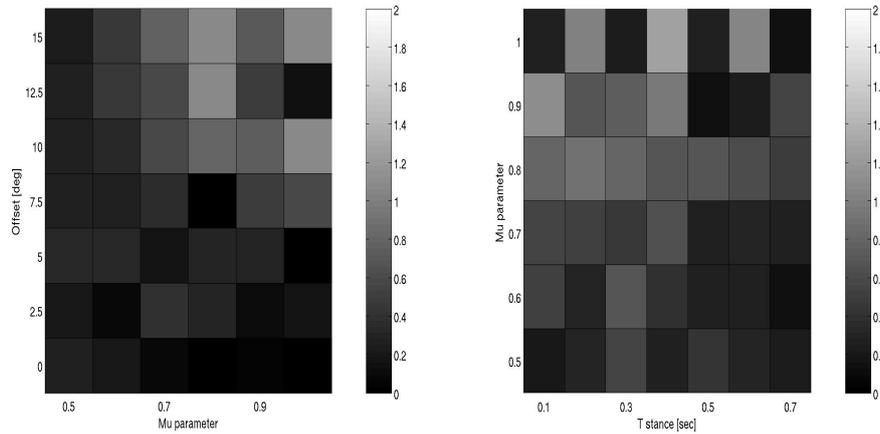


(c) *Stance duration versus offset search.*
max retraction factor fixed to 65% and
 $\sqrt{\mu}$ fixed to 0.7

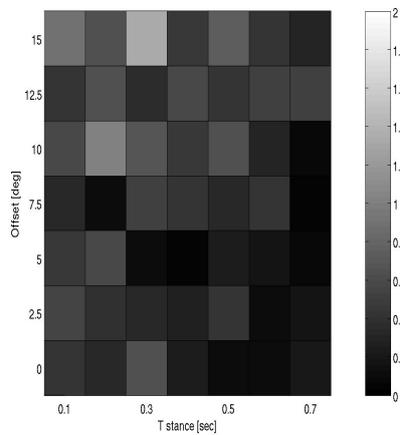
Figure 22: *Systematic searches on various parameters for the double retraction pace gait*

5.2.3 Walk gait

The systematic searches for this setup are presented in fig.23. We observe again a generally slower speeds than for the single retraction model, but less setups where the robot falls over. We observe values around $0.09m/s$ for a lot of setups. The main advantage of this gait, is that the body movements are almost non-existent, which takes the advantage observed with the single retraction model even further. Unlike other double retraction gaits, this gait can achieve comparable speeds to the single retraction gaits. We observed a speed of $0.192m/s$. Another advantage is that we can achieve great speeds with slower servos, for example at 0.6 second stance we observed speed around $0.1m/s$, which was not observed with any other gait. It should be noted, that this gait depends greatly on the starting state, the robot sometimes loses a lot of time to stabilize. Best parameter setup : 0.3 second stance phase, 15 degrees offset, 65% max retraction factor and 56 degrees oscillations.



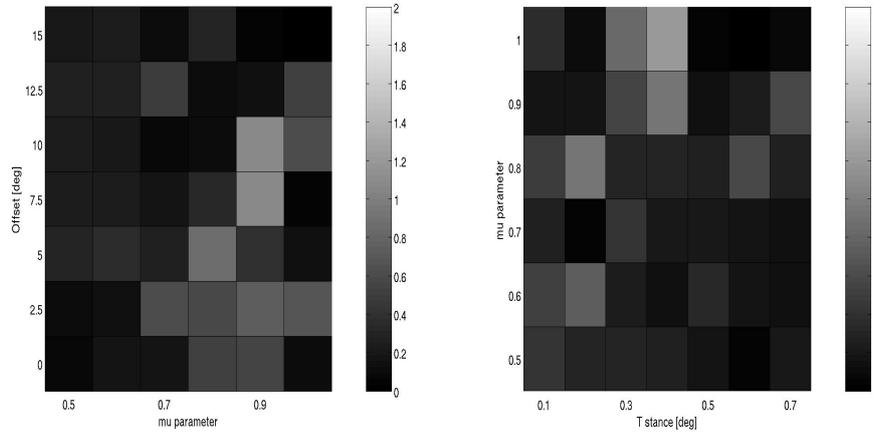
(a) *Amplitude versus Offset search.* (b) *Amplitude versus stance duration*
Stance duration fixed to 0.3 seconds and search. Offset fixed to 15 and max re-
max retraction factor fixed to 65% *traction factor fixed to 65%*



(c) *Stance duration versus offset search.*
max retraction factor fixed to 65% and
 $\sqrt{\mu}$ fixed to 0.8

Figure 23: *Systematic searches on various parameters for the double retraction walk gait*

5.2.4 Trot gait



(a) *Amplitude versus Offset search.* (b) *Amplitude versus stance duration*
Stance duration fixed to 0.3 seconds and search. Offset fixed to 5 and max retraction
max retraction factor fixed to 65% *retraction factor fixed to 65%*

Figure 24: *Systematic searches on various parameters for the double retraction trot gait*

Recall that the trot gait had convergence problems with the single retraction model. If we take a look at fig.24, we can see a much more clear pattern. The convergence problems are still present but are greatly reduced compared to the single retraction trot gait. The problem that the robot goes in circles when this gait is used remains, which is very penalizing with the metric we are using. As we can see on fig.24(a), a good amplitude of oscillation is 63 degrees, with an offset around 7.5 degrees. What is interesting, is that we managed to achieve fast speeds with slow servos ($0.17m/s$ with 0.7 second stance). Still, the searches presented here need to be redone once the convergence and direction problems are resolved. We should have also applied a different metric to this gait in order to determine the best parameter setup ignoring the direction problems. The best value achieved using this gait : $0.16m/s$. Best parameter setup : 0.3 second stance phase, 5 degrees offset, 65% max retraction factor and 70 degrees oscillations.

5.3 Conclusion

By observing the results presented above, we can conclude that the single retraction gaits are faster but are less stable. The double retraction gaits, are up to two times slower than their single retraction counterparts, but display a larger sets of usable parameters. The main issue with the double retraction model, is that we observe premature contact with the ground due to passive elements and body movements. This issue should be addressed by implementing the sensor feedback feature of the CPG, which should give a consistent boost in top speeds to the double retraction gaits. The single retraction model suffers from some instabilities due to the fact that the leg is fully extended during stance phase, which leaves the body too far from the ground. This caused the pace gait to be unusable with standard spring parameters. To address this issue, we first need to make sure that the sensory feedback is precise enough to assume that the springs are too stiff. If it the case, reducing spring stiffness produced a good pace gait with the single retraction model. To conclude this section we will attribute the best model to each gait.

1. Bounce gait : Single retraction
2. Pace gait : Single retraction if reducing the leg stiffness is possible, Double retraction otherwise
3. Walk gait : Single or Double retraction
4. Trot gait : Double retraction

6 Conclusion and Future Work

During this project, we have created a simulation model for the Cheetah robot under Webots simulation software and implemented the necessary controllers in order for it to perform different gaits. The model was calibrated to resemble as much as possible the real hardware. In order to achieve this, several hardware components had to be characterized.

The controllers that were implemented are capable of simulating different leg geometries as long as a function linking leg compression and the force applied on the leg is available as well as different control strategies like single and double retraction models discussed in chapter 4. The current model implements the three segmented leg design. Furthermore, a Central pattern generator developed in [4] was successfully implemented and tested with the

model producing four gaits. Systematic searches were performed to determine optimal setups for different gaits and compared to the data available on the real robot.

However, the model is not without flaws. Webots sensory information quality is poor, and needs to be transformed before any use. To resolve this issue, a simple filter was implemented, but it introduces a divergence with the real robot behavior. Furthermore, the sensory feedback feature of the CPG couldn't be used efficiently due to the quality of the signal. To enhance the quality of the feedback, a more complex filter should be implemented. The leg geometry could also be altered (for example adding a second touch sensor). During some experiments, we observed a sliding behavior of the feet. This issue was already mentioned by Martin Riess in his report [2], we may need to take a deeper look at friction mechanism in Webots and calibrate the friction between the feet and the ground to fit the actual robot's feet.

There are also some interesting parameters that we didn't explore in this project. As we have seen with the pace gait, we had to reduce the spring stiffness in order to have a good gait. We could perform the searches on other gaits to see if less stiff springs enhance the stability of other gait.

7 References

- 1 Webots, "http://www.cyberbotics.com" Commercial Mobile Robot Simulation Software
- 2 S. Rutishauser, *Cheetah - compliant quadruped robot*, 2008
- 3 M. Riess, *Development and Test of a Model for the Cheetah robot*, 2008
- 4 L. Righetti and A.J. Ijspeert, **Pattern generators with sensory feedback for the control of quadruped locomotion** in *Proceedings of the 2008 IEEE International conference on Robotics and Automation (ICRA 2008)*, 2008
- 5 Andre Seyfarth, Hartmut Geyer and Hugh Herr. **Swing-leg retraction: a simple control model for stable running**. *The Journal of Experimental Biology* 206, 2547-2555 (2003)
- 6 Hugh M. Herr, Gregory T. Huang and Thomas A. McMahon **A model of scale effects in mammalian quadrupedal running**. *The Journal of Experimental Biology* 205, 959-967 (2002)
- 7 L. Righetti, A. Nylén, K. Rosander, A.J. Ijspeert. Kinematics of crawling in infants. In *Submitted to the Journal of Experimental Biology*.
- 8 S. Rutishauser, Alexander Sprowitz, Ludovic Righetti and A.J. Ijspeert **Passive compliant quadruped robot using central pattern generators for locomotion control**, 2008
- 9 Alexander Sprowitz, Max Fremerey, Simon Rutishauser, Ludovic Righetti, Auke Jan Ijspeert **Compliant Quadruped Robot Cheetah Design, Implementation and Locomotion Control**, 2009
- 10 W. Schiehlen, **Energy-optimal design of walking machines**, *Multibody System Dynamics*, vol. 13, no. 1, pp 129-141, 2004.