

Exotendons for assistance of human locomotion

Fall semester project

Roderik Berthelin

Biorobotics Laboratory
Ecole Polytechnique Fédérale de Lausanne

Supervisors:
Auke Jan Ijspeert
Jesse van den Kieboom
Renaud Ronsse

January 2011

Contents

1	Introduction	3
1.1	Active assistance by powered robotics	3
1.2	Passive assistance	3
1.3	Goals	4
2	Results from the literature	4
2.1	Method	5
2.1.1	Moment optimized	6
2.1.2	Power optimized	6
2.2	Influence of the complexity of the design	7
3	Heuristic method: the PSO algorithm	8
3.1	Recap about Simulated annealing	8
3.2	PSO algorithm	9
3.3	Comparison of the two methods	11
3.3.1	Particle Swarm Optimization	11
3.3.2	Simulated annealing	11
3.4	Optimization	11
3.4.1	Moment optimized	11
3.4.2	Power optimized	13
3.4.3	Influence of the complexity of the design	13
3.5	Analysis of the results	14
3.5.1	Parameters of the algorithm	14
3.5.2	Systematic search	15
3.5.3	Initial position	16
3.5.4	Sensitivity of the configuration C, power optimized	19
4	Work with data from an adult with different speeds	23
4.1	Determination of the sampling periods of walking	23
4.2	Optimization	23
4.2.1	Slow speed	23
4.2.2	Normal speed	24
4.2.3	Fast speed	25
4.2.4	Global optimization	26
4.3	Global performance of each optimization	28
5	Simulation using Webots	30
5.1	Method	30
5.2	Optimization	31
6	Conclusion	34
7	Acknowledgments	34
8	Further Work	34
9	Bibliography	35
10	Appendix	36

1 Introduction

1.1 Active assistance by powered robotics

Powered robotic exoskeletons for assistance of human locomotion are currently under development for military and medical applications (Hal5, ALTACRO, LOKOMAT [1]).

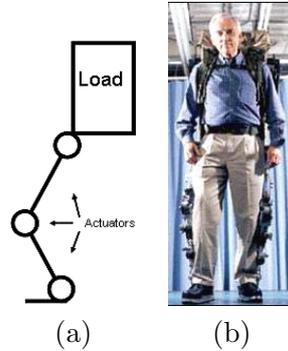


Figure 1: (a) Exoskeleton, with three actuators per leg (b) The Sarcos exoskeleton with hydraulic actuators at the joints (Huang (2004)) [2].

However, the energy requirements for such devices is important and becomes an obstacle for practical applications.

1.2 Passive assistance

It has been observed in several animals that legged locomotion is energy efficient, where the arrangement of muscles and tendons contributes to efficient locomotion. For instance, horses use long tendons in their legs to store elastic energy during running, resulting in up to a 50% decrease in energy cost per kg of body weight compared to humans.

Therefore, passive assistance systems were designed in order to reduce metabolic energy cost. Systems attached at the joints like elastic orthosis were already studied. For instance, a system working with a torsion spring placed in parallel with the knee was introduced in [3]:

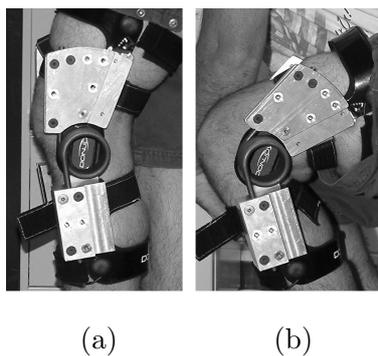


Figure 2: Prototype of elastic knee brace in (a) extended and (b) flexed positions [3].

It would result in a reduction of biological knee stiffness and muscle activation of the knee extensors.

The starting point of the study about this system was to use a torsion spring which increase the stiffness about the ankle, which would give a very different muscle activation, as showed a similar experiment. A light-weight custom-fit knee brace have been successfully designed, and can provide a rotational stiffness that is large enough to create a felt effect.

This provide information about adaptive neuromuscular behavior while hopping with knee joint

augmentation. However, for the time being, there is not enough data points to conclude about the effects on the human neuromuscular system.

The present project focused on exotendons spanning one or more joints. These exotendons would have the same behavior as traction springs and with a specific arrangement, create an exoskeleton for the human body legs.

1.3 Goals

The first objective of my work was to reproduce the results obtained in the literature regarding the optimization of different designs of the exoskeleton, and in particular the results presented in the paper "Exotendons for assistance of human locomotion" [4]. To validate the paper results with another optimization method, this project focused on the heuristic process called Particle Swarm Optimization (PSO), instead of Simulated annealing, as used in [4].

Secondly, still with PSO, the optimization was done for different gait cycle speeds in order to (i) highlight the consequences on the parameters of the exoskeleton, (ii) and possibly find a design which is robust to different gait speeds.

Finally, a simulation of the gait was done on the software Webots (<http://disal.epfl.ch/page-32502.html>) to see if the contribution of the exoskeleton would provide the active torque reduction expected, in a dynamic environment.

2 Results from the literature

The goal of this part is to overview the results obtained by van den Bogert [4] concerning conceptual exoskeletons. The author suggested a system of artificial exotendons made of a rubber-like material, with different configurations that were compared thereafter, in order to determine which one is the most efficient. An exotendon is attached with pulleys to one, three or six leg joints, depending on the configuration.

This study was focused on the sagittal plane only [3].

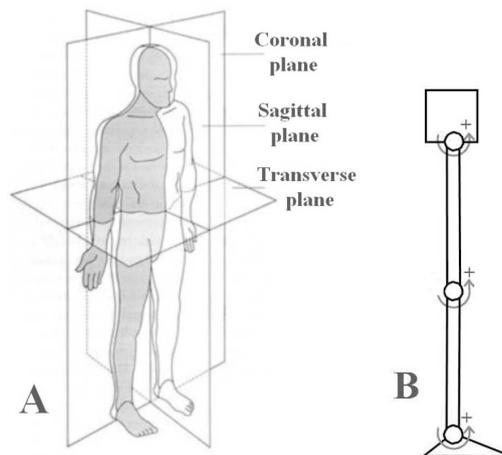


Figure 3: (a) Description of the anatomical planes and (b) Diagram of the leg shown in the rest position (0 deg at all joints) with the positive direction indicated [5].

Data were computed using a children gait [6]. Four possible exotendons configurations with increasing design complexity were considered (figure 4).

Configuration A: a one-joint exotendon in each leg, attached to the ankle. Two pulleys were necessary therefore (one pulley for each the two exotendons).

Configuration B: a three-joint exotendon in each leg, crossing the hip, knee and ankle. This design required six pulleys (three pulleys for each of the two exotendons).

Configuration C: a six-joint exotendon, crossing all joints in both legs and its twin. So we had 12 pulleys in this configuration (six pulleys for each of the two exotendons).

Configuration D: two six-joint exotendons, each crossing all joints in both legs, and their twins. 24 pulleys were required for this design (6 pulleys for each of the four exotendons).

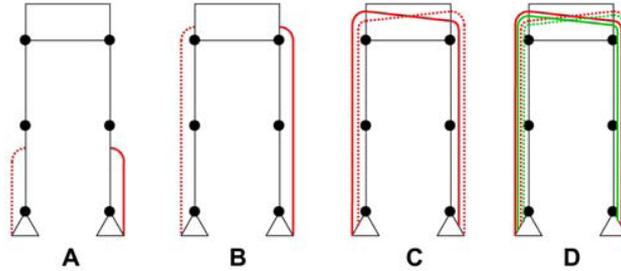


Figure 4: Possible configurations [4].

2.1 Method

Each exotendon was modeled by a spring attached at one joint (configuration A), or more joints (configurations B, C and D) with pulleys.

Exotendon length L is a linear function of six joint angles:

$$L(t) = L_0 - \sum_{i=1}^6 r_i \varphi_i(t) \quad (1)$$

where L_0 is the exotendon length when all joint angles are zero and r_i is the moment arm at joint i . The convention used is that joint angles increase during an anterior swing of the distal segment of the joint (figure 3b). Therefore, if $r_i \geq 0$, it means the exotendon runs anterior to the joint i and conversely, if $r_i < 0$, it means that the exotendon runs posteriorly (figure 4b).

The force created by an exotendon is therefore:

$$F(t) = \begin{cases} 0 & \text{if } L \leq L_{slack} \\ k[L(t) - L_{slack}] & \text{if } L > L_{slack} \end{cases} \quad (2)$$

with k the stiffness of the exotendon and L_{slack} its slack length.

So, $L(t) - L_{slack} = L_0 - \sum_{i=1}^6 r_i \varphi_i(t) - L_{slack}$, which can be written $L(t) - L_{slack} = - \sum_{i=1}^6 r_i \varphi_i(t) - (L_{slack} - L_0)$. As the slack length and L_0 are constants, we can replace their difference by one single parameter. It will be L_{slack} setting $L_0 = 0$. Thus, the moment generated at joint i is Fr_i . k takes an arbitrary value, which will be $100kNm^{-1}$. Assuming that we have N exotendons, the residual moment R_i required from the muscles crossing joint i is:

$$R_i(t) = M_i(t) - \sum_{j=1}^N r_{ij} \cdot F_j(t) \quad (3)$$

with $M_i(t)$ the total joint moments during unassisted walking, known from an inverse dynamical analysis [6].

2.1.1 Moment optimized

One of the two optimization criteria was the one based on minimization of the average residual joint moments (equation 3), over all joints and over the duration of the gait cycle:

$$C_{mom} = \frac{1}{6T} \sum_{i=1}^6 \int_0^T |R_i(t)| dt \quad (4)$$

with T the duration of the gait cycle.

The following array gives the values of the radii and slack lengths which were obtained for the different configurations, using simulated annealing as optimization method:

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
A	-60.20	N/A	N/A	N/A	N/A	N/A	6.54
B	-34.63	0.23	21.18	N/A	N/A	N/A	7.77
C	-34.31	-0.61	21.23	0	-5.63	-8.08	-4.04
D exotendon 1	-23.45	4.25	7.77	21.91	0.65	-5.24	-24.34
D exotendon 2	-15.68	-11.92	-7.60	-5.88	-4.37	10.70	-11.53

The forces and average residual moments obtained are represented on the following graphs:

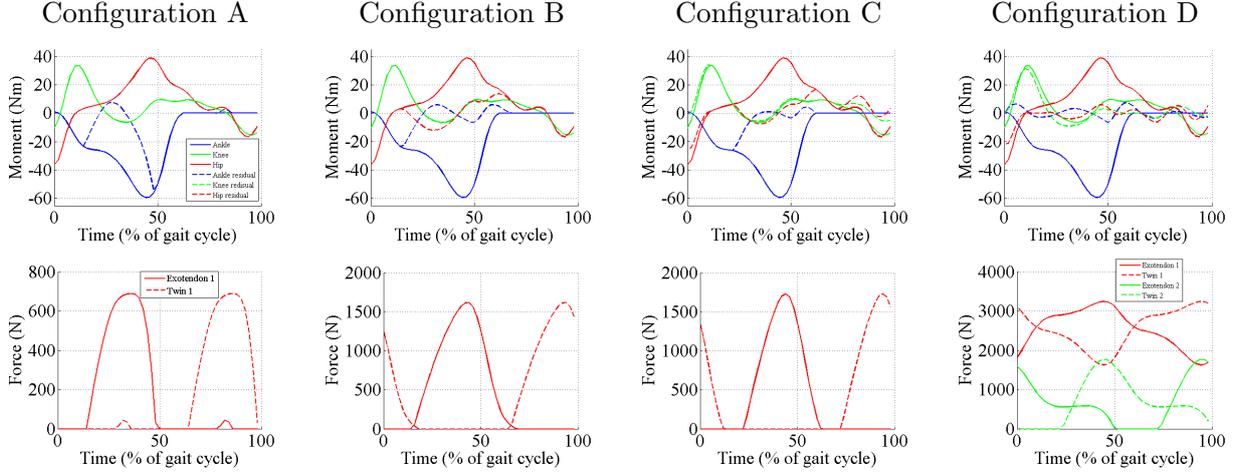


Figure 5: Moments and residual moments at the joints, and forces generated by the exotendons.

The joint moments on the right leg are shown on the top row (blue: ankle, green: knee, red: hip) with the residual moments as dashed lines. Exotendon forces are shown on the bottom row, with the twin exotendon represented by a dashed line.

2.1.2 Power optimized

The second optimization criterion was based on minimization of the average mechanical power generated by the residual moments:

$$C_{pow} = \frac{1}{6T} \sum_{i=1}^6 \int_0^T |R_i(t)\dot{\varphi}_i(t)| dt \quad (5)$$

$\dot{\varphi}_i(t)$ was calculated by analytical differentiation with a 3-point central difference method. The following array gives the values of the radii and slack lengths which were obtained for the different configurations, using the same optimization method:

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
A	-49.83	N/A	N/A	N/A	N/A	N/A	1.04
B	-32.85	12.76	33.06	N/A	N/A	N/A	-1.55
C	13.79	-2.74	-10.05	-16.12	3.22	10.87	-89.62
D exotendon 1	-19.04	-9.77	-4.69	-0.17	-5.67	10.89	-12.30
D Exotendon 2	-25.11	3.98	7.47	24.18	-2.18	-6.82	-75.33

The forces and average residual powers obtained are represented on the following graphs:

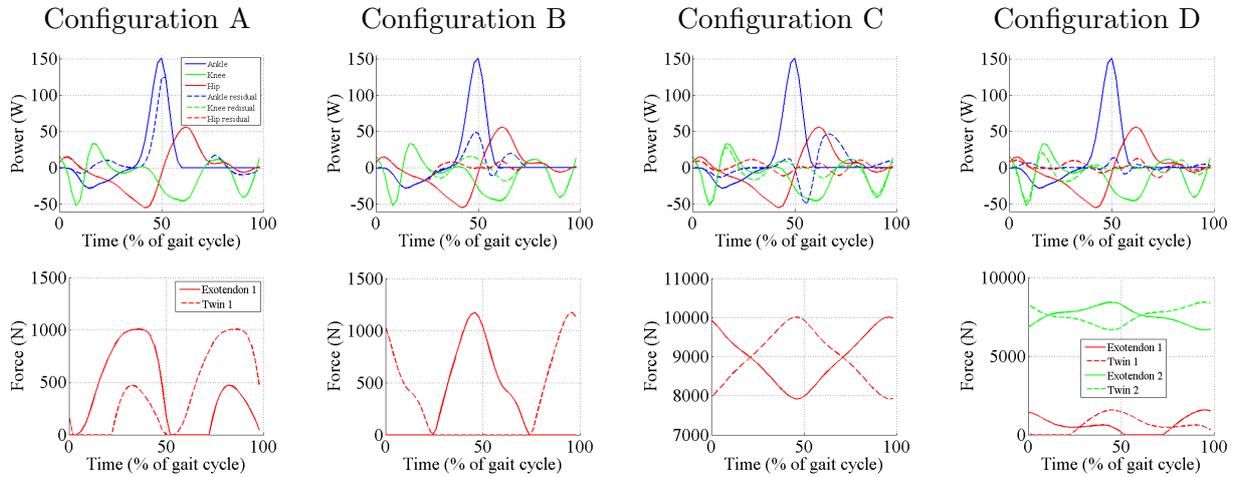


Figure 6: Moments and residual moments at the joints, and forces generated by the exotendons.

The joint powers on the right leg are shown on the top row (blue: ankle, green: knee, red: hip) with the residual powers as dashed lines. Exotendon forces are shown on the bottom row, with the twin exotendon represented by a dashed line.

2.2 Influence of the complexity of the design

The results of [4] show that the more complex the configuration, the better the minimization. Moreover, as the link between the moment and the power is $M = P\Omega$, improving one criterion also improves the other one.

Design	Moment optimized		Power optimized	
	$C_{mom}(Nm)$	$C_{pow}(W)$	$C_{mom}(Nm)$	$C_{pow}(W)$
A	11.17	18.68	11.97	18.18
B	7.56	12.26	9.06	10.64
C	7.04	12.09	7.57	9.73
D	4.08	5.66	4.37	5.18

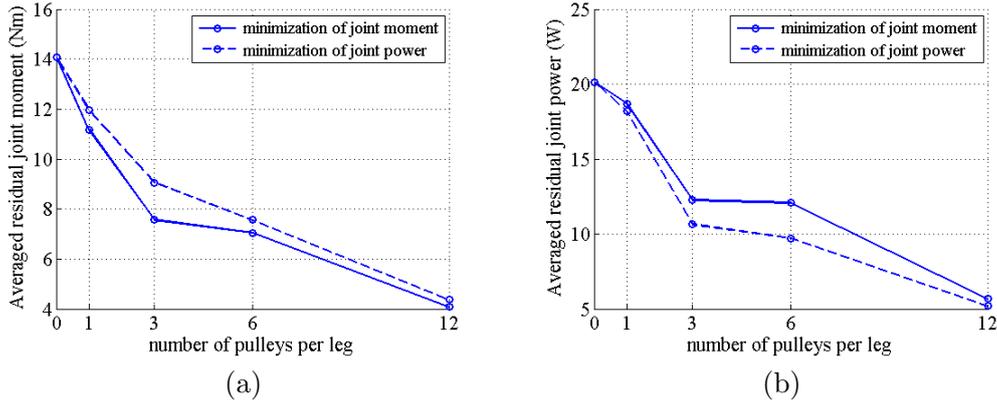


Figure 7: (a) Average residual joint moment and (b) average residual joint power according to the design.

3 Heuristic method: the PSO algorithm

To optimize the different designs with the two criteria (residual moment and residual power), van den Bogert used Simulated Annealing, a global optimization of statistical functions with simulated annealing. In this project, another heuristic process was used to figure out the best parameters for each design: the PSO (Particle Swarm Optimization) algorithm. The parameters are the slack lengths, varying between -0.3m and 0.3m , and the moment arms (radii of the pulleys), varying between -0.1m and 0.1m . The objective of this part is to put to test the same configurations as van den Bogert, but by using PSO, in order to validate the different optimizations. Alternatively, better solutions could be found with this algorithm, showing that the corresponding results reported in [4] were only local minima.

3.1 Recap about Simulated annealing

The Simulated annealing is a method which was independently described by Scott Kirkpatrick, C. Daniel Gelatt and Mario P. Vecchi in 1983, and by Vlado Cerny in 1985.

It is a generic probabilistic metaheuristic (optimization algorithm to figure out hard optimization problems) for the global optimization problem of applied mathematics, namely locating a good approximation to the global optimum of a given function in a large search space.

Metaheuristics optimize through the neighborhood approach and differ from heuristics in that they can move through neighbors that are worse solutions than the current solution.

The reason for this is that the search can't stop in a local optimum and in theory, if the metaheuristic can run for an infinite amount of time, the global optimum will be found [7].

Simulated annealing method is an algorithm based on the analogy with the thermodynamic phenomenon of the metals annealing.

In this process, first the metal is heated at a high temperature, the atoms are then agitated. Secondly, the metal is cooled progressively in order to reach a state as close as possible to the crystalline state, which is the energetic global minimum.

The algorithm sticks to the same idea: an initial high temperature is selected and an initial particle is placed in the set of solutions. Then, as the temperature decreases, the particle moves, trying to find a smaller energy state until the final temperature is reached [9].

The algorithm runs the following steps [8][9][10]:

1. Initialization: a very high starting temperature (T_0) and initial placement $m_0 \in M$, M being the finite set of solutions, are selected.
2. Movement: Randomly, a new point (m_1) is selected in the neighborhood of the original point.
3. Performance calculated: the change is computed in the performance due to the move made. Let cost function f to be optimized.
4. Choice: depending on the change in performance, accept or reject the move. The probability of acceptance depends on the current "temperature". This means that If $f(m_1) \leq f(m_0)$, m_1 is accepted as new initial placement. If a candidate m_{k+1} is worse than its predecessor m_k , this means $f(m_{k+1}) > f(m_k)$, m_{k+1} is accepted with probability p_k and is rejected with probability $1 - p_k$ where $p_k = e^{-(\Delta f/t_k)}$, t_k being the temperature at the step k (number of candidates already generated) and $\Delta f = f(m_{k+1}) - f(m_k)$. More precisely, if $p_k > r$, r being a number generated randomly in the range $[0, 1]$, the candidate becomes the new initial placement. The sequence (t_k), called cooling strategy, will be made to converge decreasingly to 0. Therefore, as the number of steps tends to infinity, the probability to accept equal deteriorations of solutions decreases and tends to 0.
5. Updating and repeating: the temperature is decreased according to a "cooling schedule" and the algorithm goes back to the step 2. The process is done until the final temperature ("Freezing Point") is reached.

Behavior of the algorithm

- The smaller Δf , the higher the acceptance probability.
- The higher t , the higher the acceptance probability.
- As t decreases, acceptance probability decreases.
- The faster t decreases, the smaller the probability to reach the global minimum.

Note

To find the best optimization of the different designs, van den Bogert set the temperature reduction rate to 1% for each 1000 cost function evaluations. The cost function was evaluated 30 million times before the algorithm stopped.

He performed the algorithm five times for each optimization to verify that the global minimum was found; the algorithm always found the same optimal design parameters.

However, he found multiple solutions for the configuration D that had nearly the same cost function values.

3.2 PSO algorithm

Particle swarm optimization (PSO) was developed by Kennedy and Eberhart [12]. It is a computation technique inspired by social behavior and movement dynamics of insects, birds and fish. It is a global gradient-less, stochastic search method. Unlike SA, in PSO, several particles are initialized through the whole space of the parameters, and each one has a random velocity. At each run of the algorithm the particles move and each one try to find a better position taking its best performance and the best performance among all the particles account. Therefore PSO uses collaboration to drive the evolutionary process.

Using details provided in "Particle Swarm Optimization, Introduction" [11], and "Eberhart and Shi, Comparing inertia weights and cons" [12], an implementation of PSO in Matlab was produced.

The algorithm runs the following steps:

1. First, a population of particles is created randomly in the parameters space. Here, this means that each particle will have initial slack lengths and moment arms representing their position. Furthermore, each particle has an initial velocity, also randomly defined.
2. Each particle position is evaluated according to the desired goal. It means that the average residual joint moments (or the average mechanical power) will be computed for each particle's position.
3. If the current position of a particle is better than its previous best position, it is updated. This means that if the particle's current position gives a smaller average residual joint moment (or average mechanical power), it will replace its previous best position.
4. Determine the best particle, i.e. the particle giving the best optimization.
5. Update particles' velocity according to the following equation:

$$v_i^{t+1} = \underbrace{w \cdot v_i(t)}_{\text{inertia}} + \underbrace{R_1(c) \cdot (p_i(t) - x_i(t))}_{\text{personal influence}} + \underbrace{R_2(c) \cdot (g(t) - x_i(t))}_{\text{social influence}} \quad (6)$$

with v_i^{t+1} the new velocity, $v_i(t)$ the previous one, $x_i(t)$ the current position, $R_1(c)$ and $R_2(c)$ random numbers in the range 0 to c , $p_i(t)$ the best previous particle's position, $g(t)$ the best previous position among all the particles and w the inertia weight, which decreases usually linearly from about 0.9 to 0.4 during a run. c was fixed at the value 1.49445 [12], which was found iteratively and gives good performances. Thus, the social and personal influences are equal.

6. Move particles to their new positions according to

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (7)$$

7. Go to step 2 until stopping criteria are satisfied, i.e. until the minimum residual moment/power is found.

To keep the average residual moment of each particle and their best performance (the minimum average residual moment they reached), a matrix composed of two rows and of the number of particles as number of columns is used: C_{mom} .

For example, below, C_{mom} has 10 particles and the one which has the best performance is the seventh one.

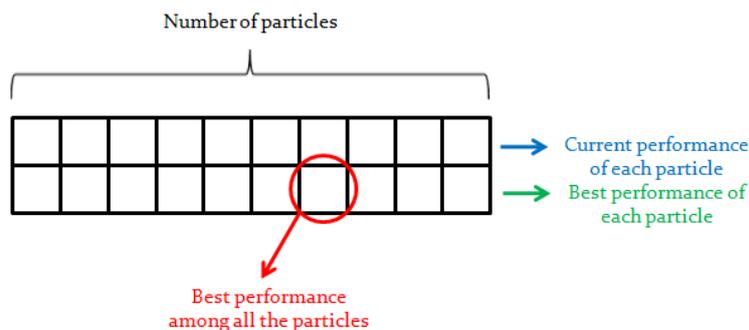


Figure 8: Matrix representing the criterion of minimization of the average residual moment .

A similar matrix to C_{mom} is used to analyse the averaged residual power: C_{pow} .

Behavior of the particles

- "Selfish": The higher the inertia weight, the more the particles follow their own path.
- "Conservative": The higher the personal influence ($R_1(c)$), the more the particles go back to the position where they got their best performance.
- "Follower": The higher the social influence ($R_2(c)$), the more the particles follow the best one of all of them (go to the position of the best performance of the best particle).

3.3 Comparison of the two methods

3.3.1 Particle Swarm Optimization

Pros

- Few algorithm parameters.
- Easy in its concept and coding implementation.
- The trajectory of each particle is influenced by their best performance but also by the collective information coming from the other trajectories.
- Low sensitivity of the algorithm (see part 3.5.4).

Cons

- Dependency on initial conditions(see part 3.5.3).
- Less efficient than Simulated Annealing to find best optimization.

3.3.2 Simulated annealing

Pros

- Ability to escape from a local minimum thanks to a neighborhood and the authorization to degrade temporarily the cost function evaluated to try to find a better position.
- Statistically guarantees finding an optimal solution [13].

Cons

- The algorithm may require large amounts of computation time.
- Deemed difficult to fine tunes to specific problems [13].

3.4 Optimization

3.4.1 Moment optimized

To bring forward the way the algorithm converges toward the minimum, the graphs, representing the average of all particles average residual momenta, the average of all particles minimum average residual momenta and the global minimum average residual moment, are represented for the configuration A, as a function of the number of iterations, by using the matrix previously introduced 8. To visualize the convergence in this configuration, 100 particles are initialized and 100 iterations are done.

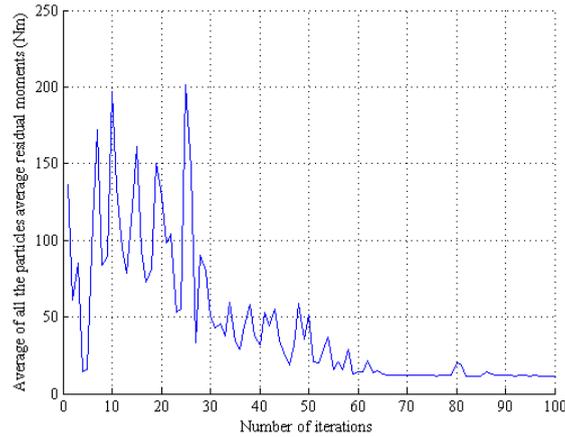
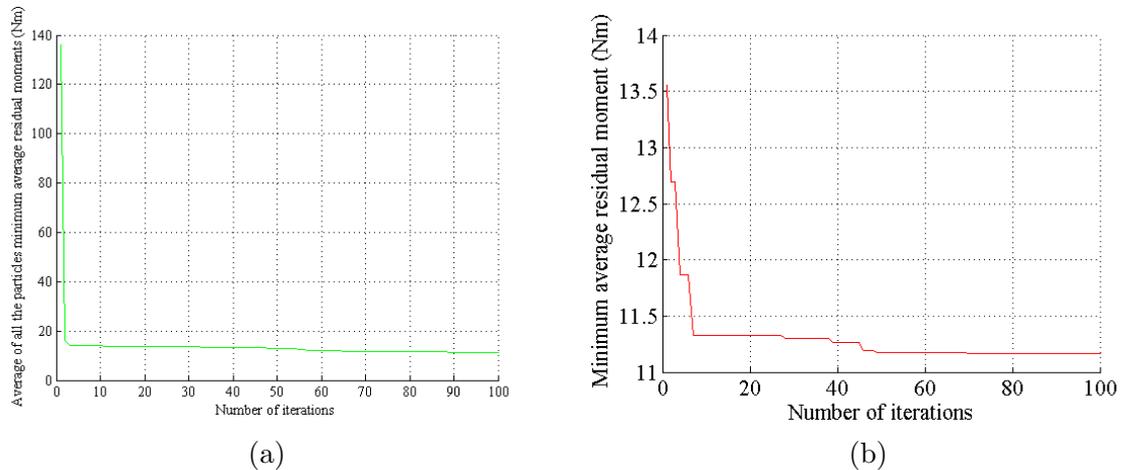


Figure 9: Average of all articles averaged residual momenta.



(a)

(b)

Figure 10: (a) Average of all particles minimum averaged residual moment and (b) minimum averaged residual moment.

At the first iteration, the average residual moment of each particle is also considered as their best performance, that is why the average of all the particles average residual momenta and the average of all the particles minimum average residual moment are equal for the first iteration. The best performance of a particle is replaced by its current average residual moment only if the latter is better than its best performance. Therefore, iteration after iteration, the particles' average residual moment can only be smaller, that's why the graph 10a is decreasing. Similarly the graph 10b is decreasing because the minimum average residual moment is replaced only if a smaller minimum is reached by a particle.

The first graph shows that the average of all the particles' averaged residual momenta can be worse after a bigger number of iterations. For example, after 39 iterations, the average was about 70Nm whereas at the tenth iteration, it was about 15Nm. That means that even if the best performances of the particles are overall better and better (figure 10a) during the run of the algorithm, some of them go through by positions where their performance can be worse than previously. The movements of the particles will be discussed in the section 3.5.3.

I was able to converge to the same minimum as reported in [4] for all the configurations except the configuration D.

These are the optimizations obtained:

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
A	-60.20	N/A	N/A	N/A	N/A	N/A	6.54
B	-34.63	0.23	21.18	N/A	N/A	N/A	7.77
C	-34.31	-0.61	21.23	0	-5.63	-8.08	-4.04
D exotendon 1	0	-7.36	-8.16	-34.01	1.29	21.05	-4.38
D exotendon 2	1.65	12.31	13.34	-29.63	7.31	4.80	4.11

3.4.2 Power optimized

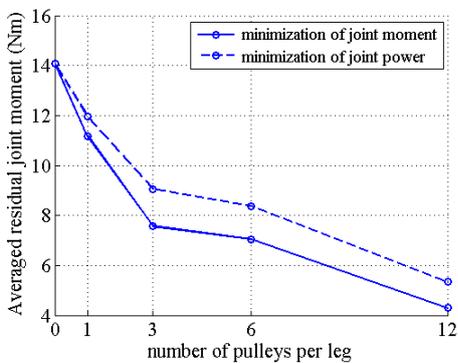
I was able to converge to the same minimum than reported in [4] for the configurations A and B but not for the C and D. For the configuration C, a sensitivity analysis was done to understand why the PSO algorithm did not find the same minimum (see part 3.5.4).

These are the optimizations obtained:

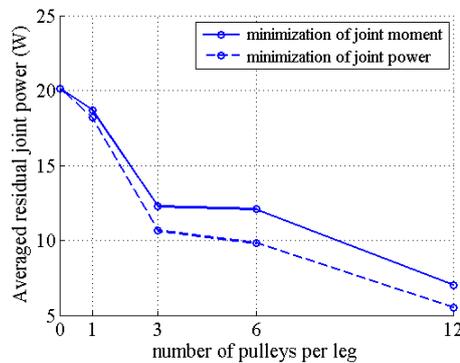
Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
A	-49.83	N/A	N/A	N/A	N/A	N/A	1.04
B	-32.85	12.76	33.06	N/A	N/A	N/A	-1.55
C	14.74	-2.24	-8.59	-21.97	4.39	11.91	-20.25
D exotendon 1	-24.03	17.55	4.47	0.12	12.90	25.05	7.20
D Exotendon 2	0	-8.94	-10.87	-42.09	10.07	27.65	3.43

3.4.3 Influence of the complexity of the design

Design	Moment optimized		Power optimized	
	$C_{mom}(Nm)$	$C_{pow}(W)$	$C_{mom}(Nm)$	$C_{pow}(W)$
A	11.17	18.68	11.97	18.18
B	7.56	12.26	9.06	10.64
C	7.04	12.09	8.37	9.82
D	4.29	7.02	5.34	5.52



(a)



(b)

Figure 11: (a) Average residual joint moment and (b) average residual joint power according to the design.

3.5 Analysis of the results

3.5.1 Parameters of the algorithm

During the test of the algorithm, I couldn't find the minimum of the residual joint moment/power each time I ran the program. To understand why the algorithm does not converge systematically, the different parameters have to be analyzed:

1. The number of particles, that means the number of complex containing the radii and the slack lengths.
2. The number of parameters defining a position (two for the configuration A for example, the slack length and the radius of the pulley of the ankle)
3. The number of iterations of the algorithm.
4. The values of the parameters of the equation updating the velocity: the inertia and the range of the random numbers.

To understand the influence of the number of particles and the number of iterations, we can display the minimum average residual moment/power as a function of the number of particles and iterations. Here, the algorithm is run each time for a new couple (number of particles ; number of iterations). The following graphs show the results for the configuration A for the moment minimization criterion. As the PSO is based on random numbers generation, the graphs obtained are only examples and therefore give only the trends of the convergence. Other simulations were done and gave similar graphs.

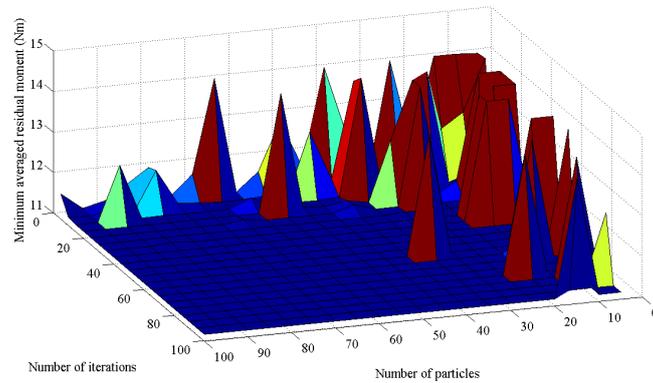


Figure 12: Minimum average residual moment depending on the numbers of iterations and particles.

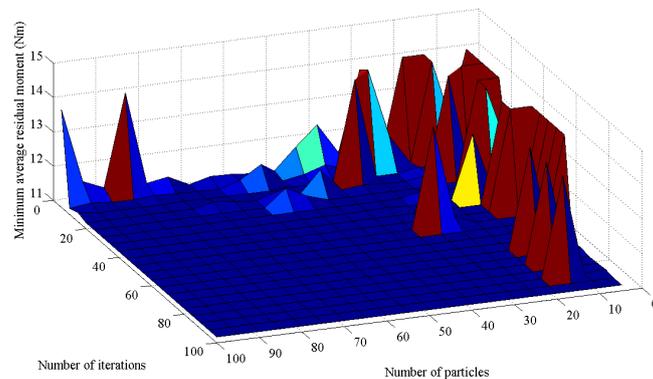


Figure 13: Minimum average residual moment depending on the numbers of iterations and particles.

We can see for example on the figure 12 that, with 20 particles, the convergence was better with 25 iterations (11.17Nm) than 80 (14.09Nm) (whereas with more iterations, the particles would be supposed to have more "time" to reach the position of the minimum average residual moment). We can observe the same phenomenon in the other way: with 15 iterations, the convergence was better with 40 (11.20Nm) particles than 90 (12.54Nm) (whereas with more particles, the convergence would be supposed to be better thanks to a better social influence). As the inertia, the range of the random numbers and the number of parameters defining a position are the same, there is another parameter which influences the convergence, and which has to be random. Obviously, as the velocity of each particle is updated randomly with the two coefficients $R_1(c)$ and $R_2(c)$, it could be a reason for which the particles converge differently. Importantly the initial position of the particles, which is also randomized, has an influence on the algorithm behavior.

To highlight the consequences of the initialization, we can study the movements of some particles which would have been initialized in areas in which the fitness function landscape is different (see part 3.5.3).

3.5.2 Systematic search

A systematic search of the minimum average residual moment in function of the position of the particles was done in order to observe their behavior. As the configuration A has only 2 parameters for the position of a particle (one slack length and one radius), it is easy to represent the systematic search in this case with a 3D graph. It shows the minimum average residual moment depending on the slack length (varying between -0.3 and 0.3) and the radius of the pulley of the ankle (varying between -0.1 and 0.1).

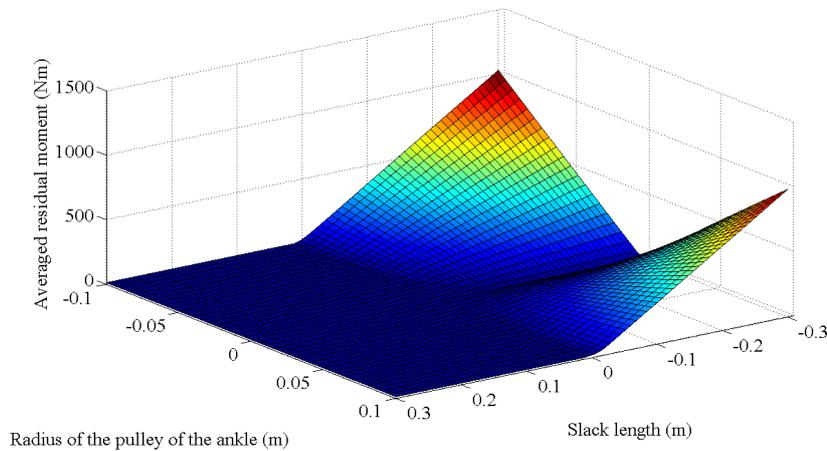


Figure 14: Systematic search of the average residual moment for the design A.

We can notice significant differences depending on the couple (slack length ; radius) which give different tendencies to the graph. To understand the curvatures, we can explore what is induced by the variation of the two parameters.

First, we can see that for a positive slack length, the average residual moment is flat. Indeed, since $0 \leq L_{slack} \leq 0.3$, it will be more likely to have $F(t) = 0(L \leq L_{slack})$ than $F(t) = k[L(t) - L_{slack}](L > L_{slack})$ (equation 2) especially when L_{slack} increases. We can notice this on the following figure which represents the average force of the exotendon in function of the radius and the slack length:

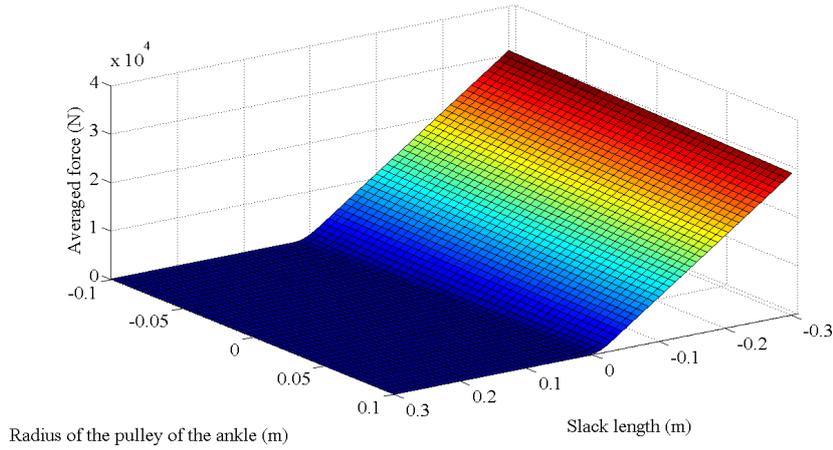


Figure 15: Systematic search of the average force for the design A.

Therefore, we will probably have $R(t) = M(t)$ (equation 3), which is the residual joint moment without contribution of the exotendons, 14.09Nm.

After, when the slack length decreases from 0 to -0.3, it will be more likely to have $F(t) = k[L(t) - L_{slack}] (L > L_{slack})$ and which will increase (figure 15). Moreover, the larger the radius, the larger $|rF|$. Thus, $|R(t) = M(t) - rF|$ will increase and therefore the residual average moment (equation 4) as well. This explains the two slopes in the negative part of the slack length (figure 14).

Finally, when $r \simeq 0$, $L(t) = -r\varphi(t) \simeq 0$, so $R(t) = M(t) - rF \simeq R(t) = 14.09Nm$. This result explains the hollow between the two slopes (figure 14).

Below we can see the area around the minimum average residual moment, represented by a pink spot (radius = -60.2mm and slack length = 6.54mm).

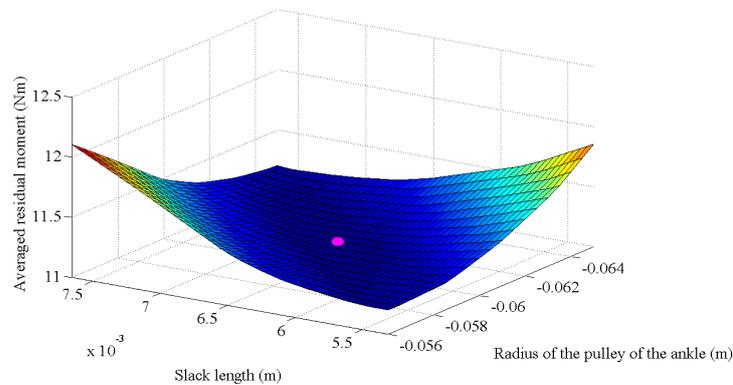


Figure 16: Systematic search of the average residual moment for the design A around the minimum.

3.5.3 Initial position

To examine the movements of the particles depending on their initial position, we recorded the behavior of some particles if they were placed:

1. far from the minimum in a flat area (in black)
2. in a steep area (in green)
3. close to the minimum in a flat area (in purple)
4. around the minimum (in red)

Theses places are indicated on the following graph, the white circle representing the minimum:

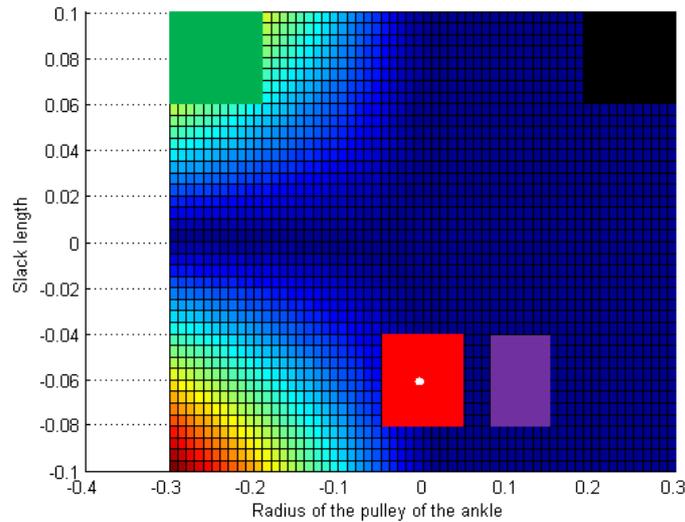


Figure 17: Initial areas of the particles.

On the graphs representing the movements of the particles, the red circle represents the position for which the average residual moment is minimum and the diamonds are the initial positions of the particles. Each way of a particle is represented by one color.

For clarity of the graphs, the positions of the particles are not displayed for each iteration.

The videos sequences are available on the website <http://biorob.epfl.ch/page-53599.html>.

Particles initialized far from the minimum in a flat area

For this case, 21 particles were initialized with a radius between 0.06 and 0.1m and a slack length between 0.2 and 0.3m.

The locations of the particles are displayed at the first iteration and every 400 iterations until the 2000th.

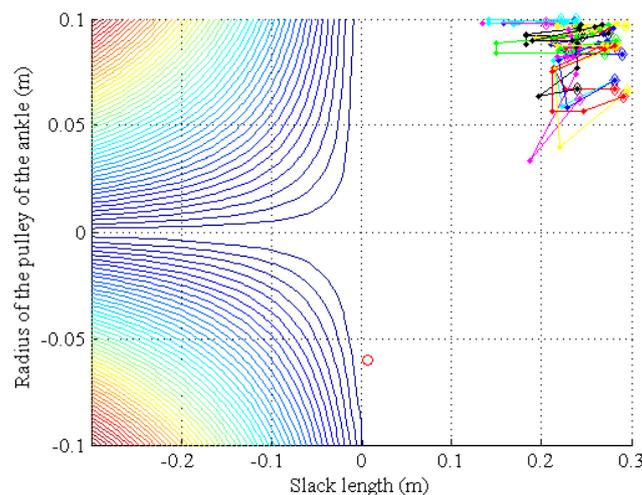


Figure 18: Particles initialized far from the minimum, in a flat area.

The minimum was almost never reached, with an optimal performance of 14.09Nm (the performance without contribution of the exotendons).

Particles initialized in a steep area

For this case, 7 particles are initialized with a radius between 0.06 and 0.1m and a slack length between -0.3 and -0.2m.

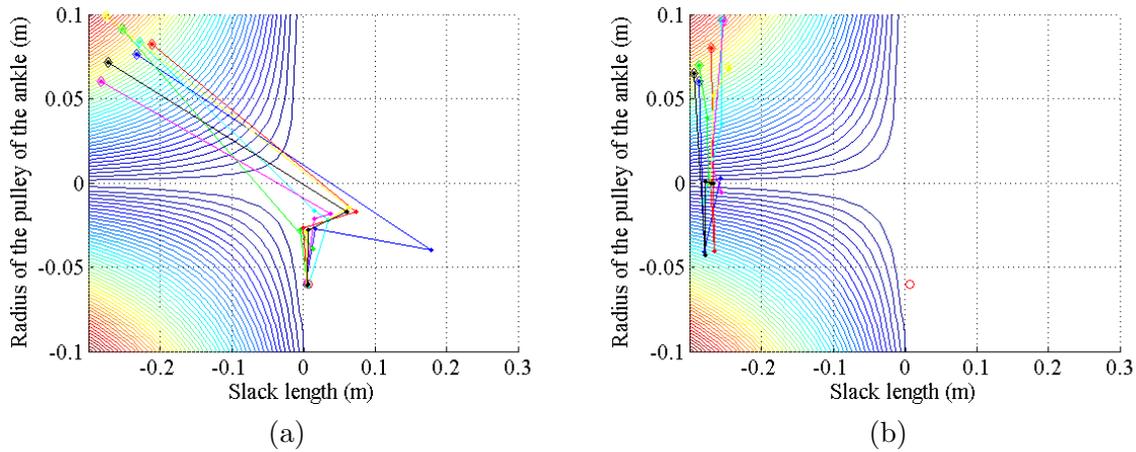


Figure 19: Particles initialized in a steep area and (a) having reached the minimum (b) having not reached the minimum.

Sometimes, the particles reached the minimum (figure 19a), but sometimes they got stuck, for example between the two slopes(figure 19b); here their best performance was 13,84Nm.

Particles initialized close to the minimum in a flat area

For this case, 7 particles are initialized with a radius between -0.08 and -0.04m and a slack length between 0.075 and 0.15m; 100 iterations were done. The locations of the particles are displayed at the first iteration and every 20 iterations until the 100th.

The minimum was often reached (figure 20a).

Particles initialized around the minimum

For this case, 7 particles are initialized with a radius between -0.08 and -0.04m and a slack length between -0.05 and 0.05m.

The locations of the particles are displayed at the first iteration and every 20 iterations until the 100th.

The particles always reached the minimum (figure 20b).

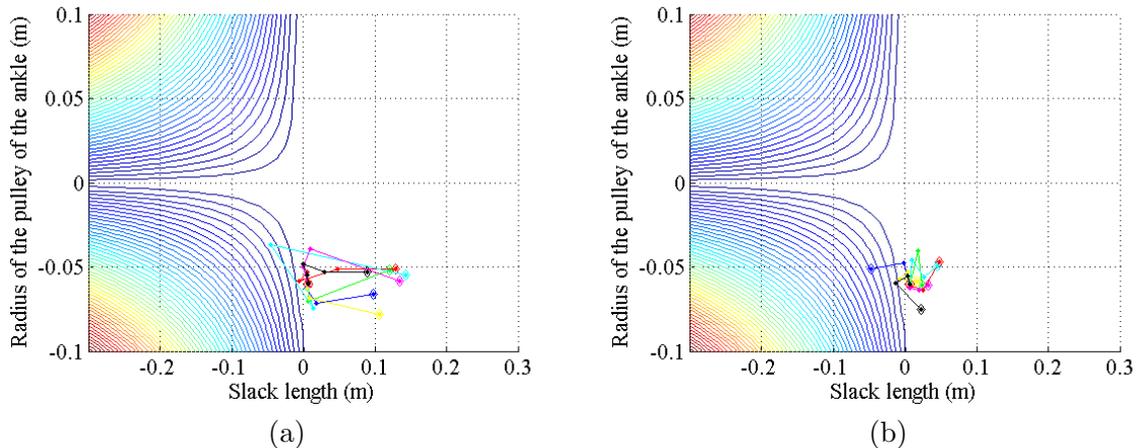


Figure 20: Particles initialized (a) close to the minimum and (b) around the minimum.

Interpretation

It's true that the more there are particles, the better the social influence as the probability that some particles goes to the minimum increases. After, the more the number of iterations, the more the particles have time to reach the minimum.

However, in some cases the particles get stuck, as shown on figure 18, because in this area, the average residual moment has no clear gradient.

In the case where the particles were on a steep place, they get off it quickly as the gradient is very high, to reach an area where the average residual moment is much smaller.

Finally, in the case where the particles are initialized around the minimum, it's easier for them to reach the minimum.

Therefore, the initial position of the particles is important, and, more generally, the landscape of the area where the particles move. Even though they are few, if one of them is close enough to the minimum, it will be easier for them to reach it in a limited number of iterations. But they can be unsuccessful if they got stuck in a local minimum, or in a flat area, where the average residual moment is almost unvarying.

Regarding the part 3.4.1, we saw that the average of all particles average residual moment is not necessarily decreasing. This can be explained by the fact that to reach a better position, the particles can go through by local extrema, placed between the better position and the current one, which increase their average residual moment.

3.5.4 Sensitivity of the configuration C, power optimized

For the design C, as said previously, the algorithm was not able to converge to the same optimization than the one obtained in the literature for the average residual power.

Most of the time, the best result got was 10.27W and once the result was 9.82W (whereas the minimum would be 9.73W).

To understand why it is difficult for PSO to find the best optimization, a sensitivity analysis of this case was done.

Here, the sensitivity is the variation of the average residual power when the value of one parameter (slack length or radius of a pulley) change:

$$S = \frac{\frac{\Delta P}{P}}{\frac{\Delta p}{p}} = \frac{\frac{P - P_{opt}}{P_{opt}}}{\frac{p - p_{opt}}{p_{opt}}} \quad (8)$$

with P_{opt} the average residual power and p_{opt} the value of the parameter, both obtained with the considered optimization and P the average residual power after the change of the parameter and p the new value of the parameter.

However, here, we simply displayed the average residual power after the change of parameters.

The range of variation of the parameters is $\frac{p - p_{opt}}{p_{opt}} = \pm 0.1$ which means a range of variation of $\pm 10\%$. The analysis was done in the range [9.7W;14.5W].

van den Bogert's best configuration: $C_{pow} = 9.73W$

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
	13.79	-2.74	-10.05	-16.12	3.22	10.87	-89.62

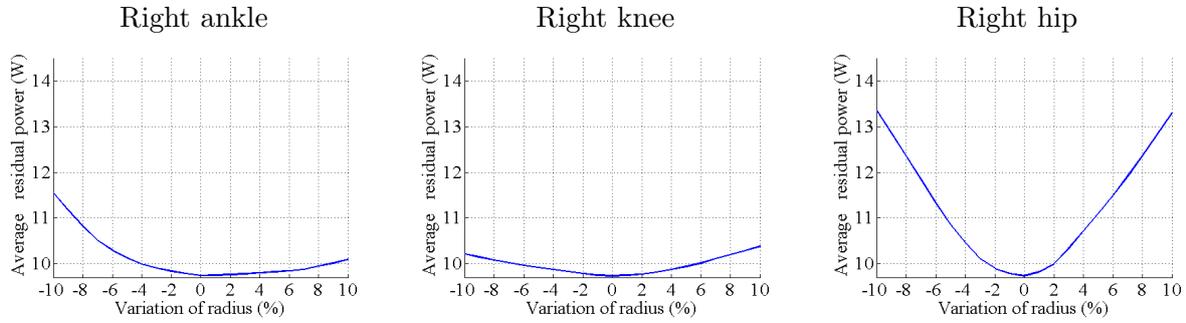


Figure 21: Influence of the radii of the pulleys of the right leg on the average residual power.

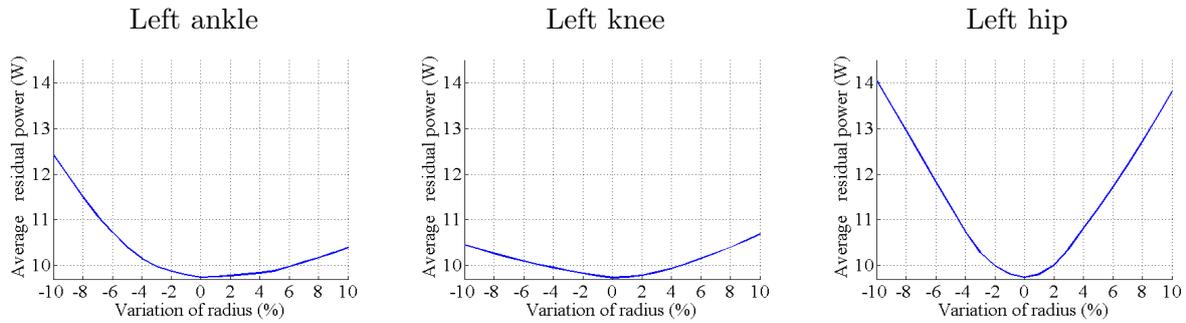


Figure 22: Influence of the radii of the pulleys of the left leg on the average residual power.

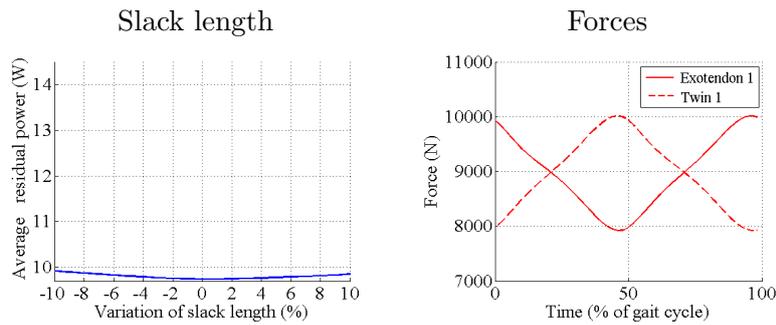


Figure 23: Influence of the slack length on the average residual power and forces generated of the exotendons.

My own best solution: $C_{pow} = 9.82W$

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
	14.74	-2.24	-8.59	-21.97	4.39	11.91	-20.25

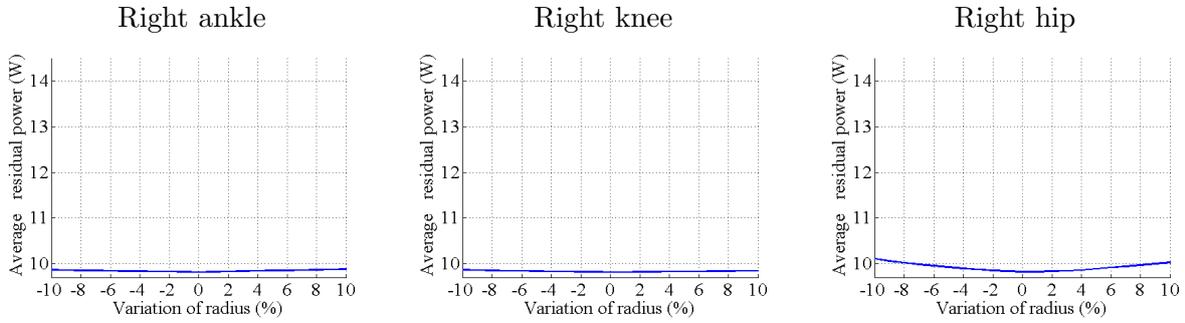


Figure 24: Influence of the radii of the pulleys of the right leg on the average residual power.

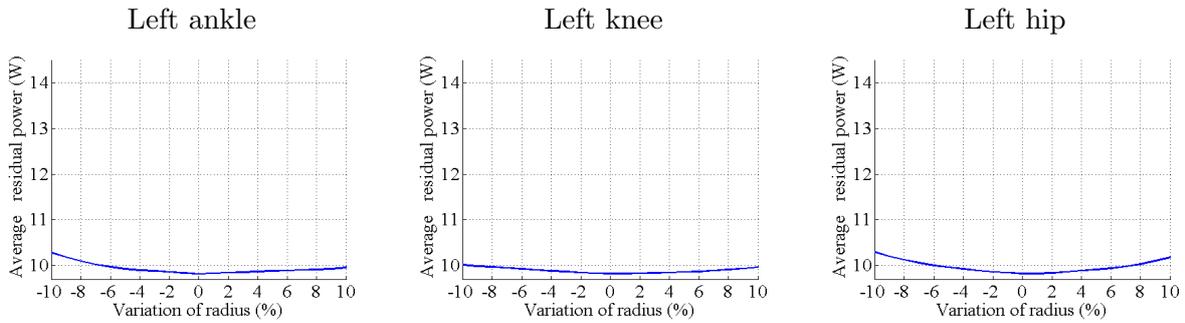


Figure 25: Influence of the radii of the pulleys of the left leg on the average residual power.

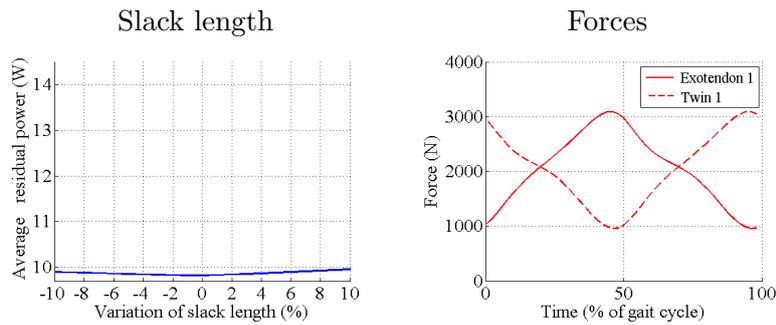


Figure 26: Influence of the slack length on the average residual power and forces generated by the exotendons.

The most attractive solution: $C_{pow} = 10.27W$

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
	0	-4.56	-4.79	-32.02	11.04	29.38	-1.27

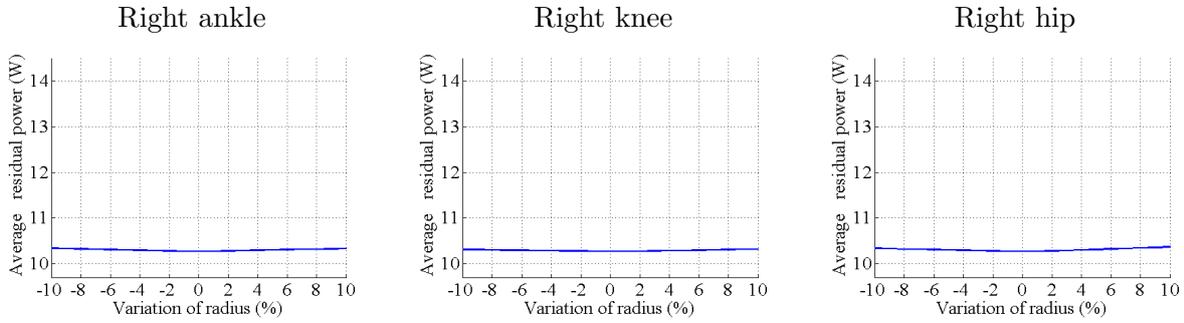


Figure 27: Influence of the radii of the pulleys of the right leg on the average residual power.

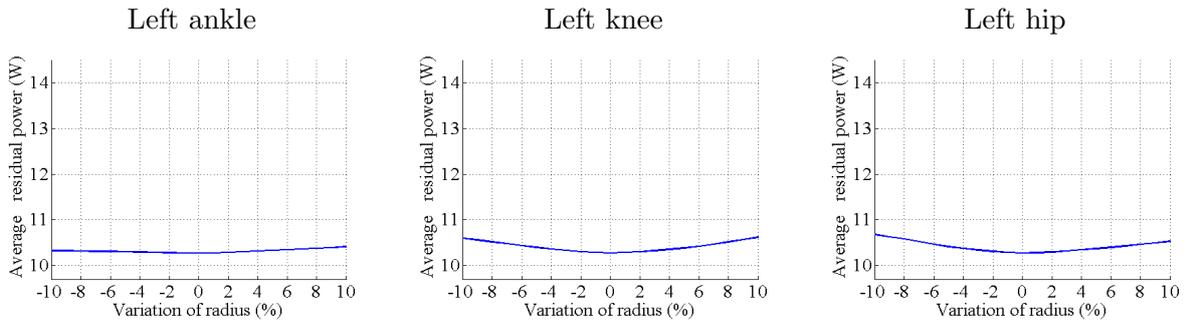


Figure 28: Influence of the radii of the pulleys of the left leg on the average residual power.

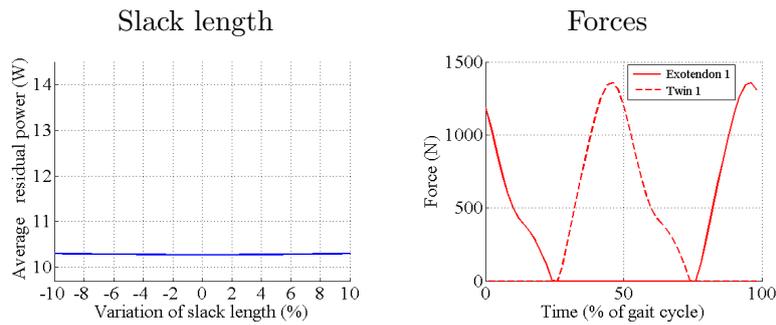


Figure 29: Influence of the slack length on the average residual power and forces generated by the exotendons.

Interpretation

We see that the worse the minimization, the less sensitive the optimization.

This can be understood by observing that more sensitive solutions corresponded to smaller slack lengths, and therefore, to higher forces provided by the exotendons (figure 15). Thus, very likely, the system might become counterproductive, i.e. giving rise to the opposite effect as the energy (see part 4.3, configuration C).

Also we see that the parameter changes involve the highest change of average residual power for the best optimization.

This analysis shows that PSO tends to converge to solutions which are less sensitive to parameter changes, and therefore, which are more robust. This would be more useful in the case of an implementation of such exoskeleton, as a counterproductive system would make no sense.

4 Work with data from an adult with different speeds

The objective of this part is to compute and compare similar optimizations performed for different speeds.

To do this, we used the angles, moments and powers for the joints of an adult whose weight was 70 kilos. These data were found in the literature for three different gait speeds: slow, normal and fast [14].

4.1 Determination of the sampling periods of walking

The times separating the successive angular positions of the joints for the different speeds were not provided.

To determine them, the least squares approximation was used (see appendix). More precisely it's the time Δ_{moy} separating three successive angular positions which was determined.

The following results were obtained:

$$\begin{aligned}\Delta_{moy,slow} &\simeq 53.8ms \\ \Delta_{moy,normal} &\simeq 44.9ms \\ \Delta_{moy,fast} &\simeq 39.1ms\end{aligned}$$

The least square method was computed taking the three joints into account.

4.2 Optimization

4.2.1 Slow speed

Without contribution of the exotendons, the moment at the joints is 16.67Nm and the power 12.51W. Using the same algorithm as before (PSO), the optimizations for the moment and the power give the following results:

Moment optimized

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
A	-83.56	N/A	N/A	N/A	N/A	N/A	0.17
B	-68.59	-3.93	9.55	N/A	N/A	N/A	-3.49
C	-59.63	-5.30	7.99	6.30	-3.94	-5.08	-7.75
D exotendon 1	1.00	3.48	-26.56	-24.48	27.90	31.61	20.49
D exotendon 2	-59.25	-6.76	6.79	8.11	-4.42	-4.12	-8.55

Power optimized

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
A	-75.76	N/A	N/A	N/A	N/A	N/A	-1.77
B	-61.65	-0.67	17.66	N/A	N/A	N/A	-5.48
C	4.58	-3.83	-3.74	-62.21	-0.94	9.95	-6.77
D exotendon 1	-60.60	-4.50	6.21	8.43	-4.92	-4.59	-7.67
D exotendon 2	-21.32	20.58	36.95	-11.36	8.66	-4.02	13.20

Comparison

Design	Moment optimized		Power optimized	
	$C_{mom}(Nm)$	$C_{pow}(W)$	$C_{mom}(Nm)$	$C_{pow}(W)$
A	8.29	9.03	8.93	8.93
B	6.59	7.82	7.87	7.46
C	5.65	6.13	6.13	5.70
D	3.83	3.55	4.45	3.35

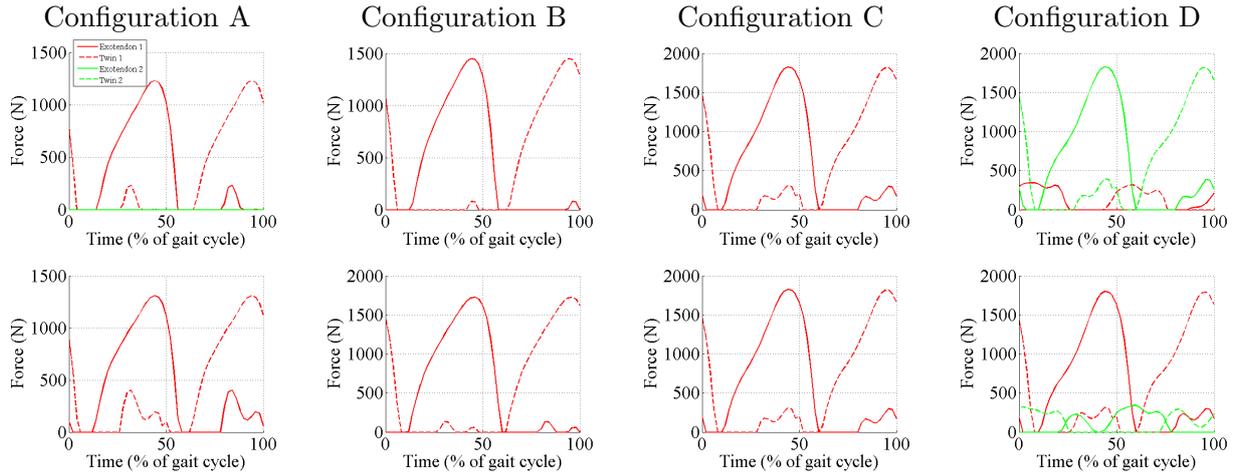
Forces

Figure 30: Forces, first row: moment optimized, second row: power optimized. The twin exotendons are represented by a dashed line.

4.2.2 Normal speed

Without contribution of the exotendons, the moment at the joints is 19.44Nm and the power 20.88W. The optimizations for the moment and the power give the following results:

Moment optimized

Leg	Right			Left			L_{slack}
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	
A	-76.04	N/A	N/A	N/A	N/A	N/A	1.23
B	-59.62	-1.78	13.95	N/A	N/A	N/A	-3.87
C	-50.46	-0.65	9.94	8.42	-7.10	-5.97	-10.12
D exotendon 1	-6.61	3.30	-33.85	-8.92	21.61	33.74	21.54
D exotendon 2	8.93	-7.57	-5.79	-51.00	-2.57	8.85	-10.46

Power optimized

Leg	Right			Left			L_{slack}
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	
A	-78.94	N/A	N/A	N/A	N/A	N/A	-1.27
B	-56.53	2.31	20.84	N/A	N/A	N/A	-6.85
C	5.96	-6.71	-3.63	-56.72	3.59	10.85	-8.13
D exotendon 1	-56.62	0.74	9.21	7.84	-7.55	-3.00	-9.35
D exotendon 2	-11.50	23.41	37.03	-6.54	9.57	-28.30	23.88

Comparison

Design	Moment optimized		Power optimized	
	$C_{mom}(Nm)$	$C_{pow}(W)$	$C_{mom}(Nm)$	$C_{pow}(W)$
A	12.04	17.14	13.65	16.61
B	10.32	14.92	12.02	14.02
C	8.33	11.65	9.10	10.80
D	5.93	7.44	6.89	6.99

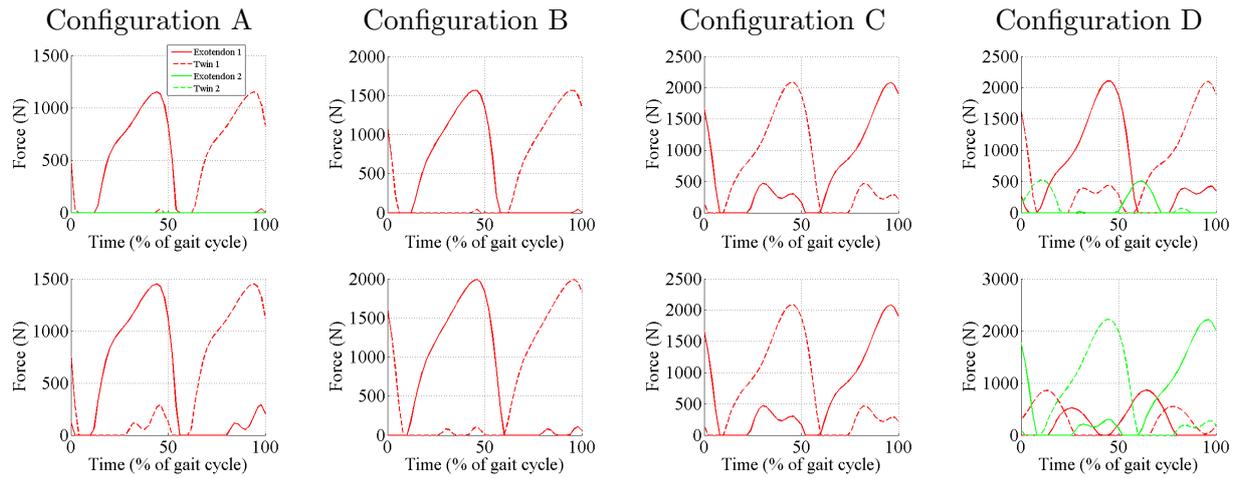
Forces

Figure 31: Forces, first row: moment optimized, second row: power optimized. The twin exotendons are represented by a dashed line.

4.2.3 Fast speed

Without contribution of the exotendons, the moment at the joints is 24.37Nm and the power 33.82W. The optimizations for the moment and the power give the following results:

Moment optimized

Leg	Right			Left			L_{slack}
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	
A	-78.07	N/A	N/A	N/A	N/A	N/A	3.19
B	-40.02	3.63	28.61	N/A	N/A	N/A	-6.72
C	6.73	-8.66	-10.10	-38.50	2.89	21.36	-12.47
D exotendon 1	-76.04	-4.12	26.23	-0.51	-13.88	-15.14	1.23
D exotendon 2	-53.46	43.22	74.57	19.31	13.78	3.16	27.84

Power optimized

Leg	Right			Left			L_{slack}
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	
A	-89.83	N/A	N/A	N/A	N/A	N/A	2.45
B	-50.69	7.42	31.46	N/A	N/A	N/A	-4.64
C	23.59	-7.89	-13.88	-29.31	10.20	19.61	-21.30
D exotendon 1	-76.04	-4.71	22.51	-2.99	-14.30	-16.47	1.23
D exotendon 2	-37.90	31.47	62.32	15.47	9.94	6.23	18.04

Comparison

Design	Moment optimized		Power optimized	
	$C_{mom}(Nm)$	$C_{pow}(W)$	$C_{mom}(Nm)$	$C_{pow}(W)$
A	19.64	32.52	20.58	32.23
B	14.36	23.91	14.94	23.14
C	12.09	21.82	14.58	18.61
D	9.94	13.53	10.21	13.01

Forces

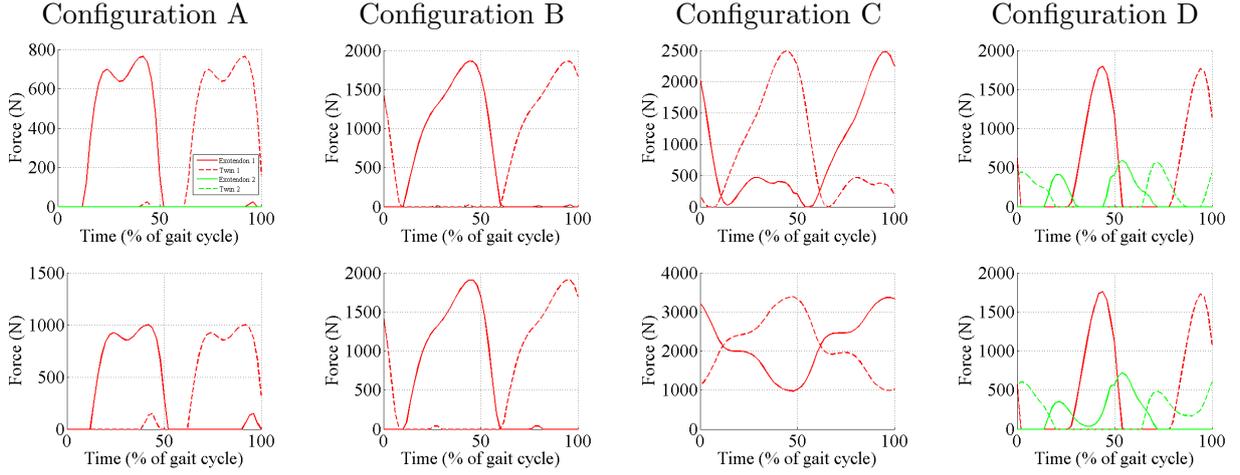


Figure 32: Forces, first row: moment optimized, second row: power optimized. The twin extensors are represented by a dashed line.

4.2.4 Global optimization

Here, we looked for a configuration that would be a sort of compromise between the three speeds. So the criteria of minimization were $C_{mom} = \frac{1}{3}(C_{mom,slow} + C_{mom,normal} + C_{mom,fast})$ and $C_{pow} = \frac{1}{3}(C_{pow,slow} + C_{pow,normal} + C_{pow,fast})$.

Moment optimized

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
A	-85.94	N/A	N/A	N/A	N/A	N/A	2.11
B	-60.37	-1.40	17.82	N/A	N/A	N/A	-4.70
C	5.29	-6.11	-6.92	-50.62	-0.80	13.51	-9.10
D exotendon 1	8.09	-8.02	-8.86	-49.46	-3.05	9.86	-10.35
D exotendon 2	-8.29	7.02	-6.46	-2.49	14.18	18.93	7.55

Power optimized

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
A	75.68	N/A	N/A	N/A	N/A	N/A	-1.49
B	-55.82	3.17	22.90	N/A	N/A	N/A	-5.29
C	-55.84	3.94	17.28	2.33	-5.22	-5.32	-6.43
D exotendon 1	-2.23	13.25	22.56	-6.08	9.56	-5.54	8.13
D exotendon 2	6.61	-7.17	-8.97	-53.36	-0.98	10.38	-9.09

Comparison

Design	Moment optimized		Power optimized	
	$C_{mom}(Nm)$	$C_{pow}(W)$	$C_{mom}(Nm)$	$C_{pow}(W)$
A	14.02	19.68	14.84	19.36
B	11.47	16.24	12.04	15.53
C	10.26	14.13	10.86	13.44
D	8.30	10.03	8.76	9.46

The forces which would be created by the exotendons with these optimizations at the different speeds are:

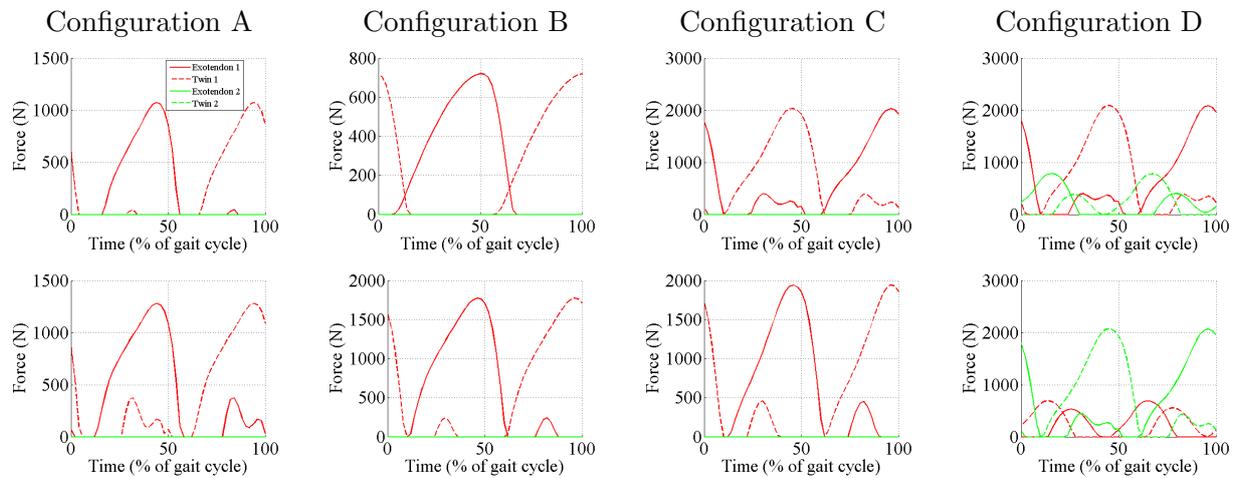
Forces, slow speed

Figure 33: Forces, first row: moment optimized, second row: power optimized. The twin exotendons are represented by a dashed line.

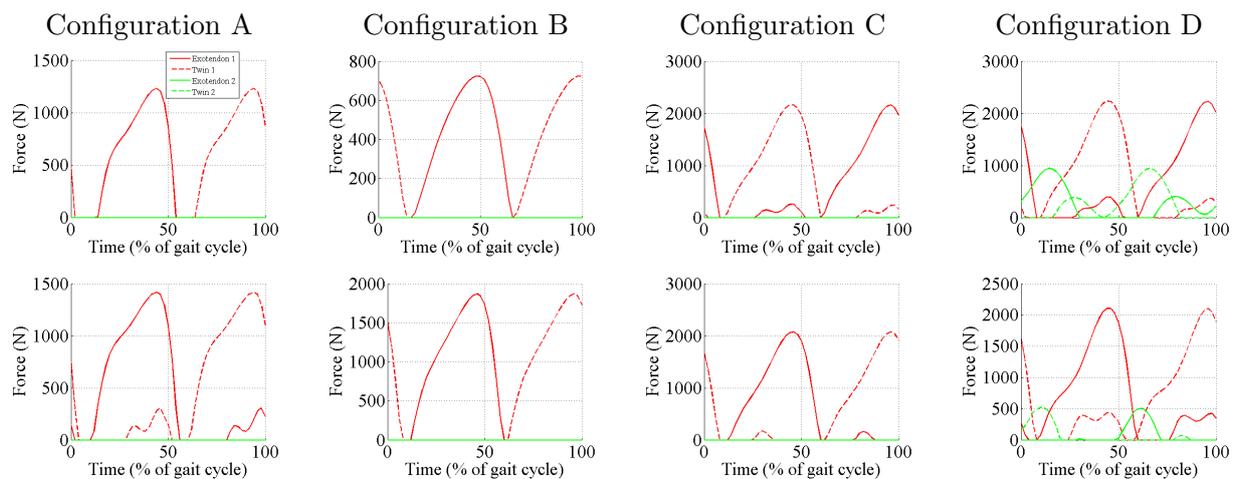
Forces, normal speed

Figure 34: Forces, first row: moment optimized, second row: power optimized. The twin exotendons are represented by a dashed line.

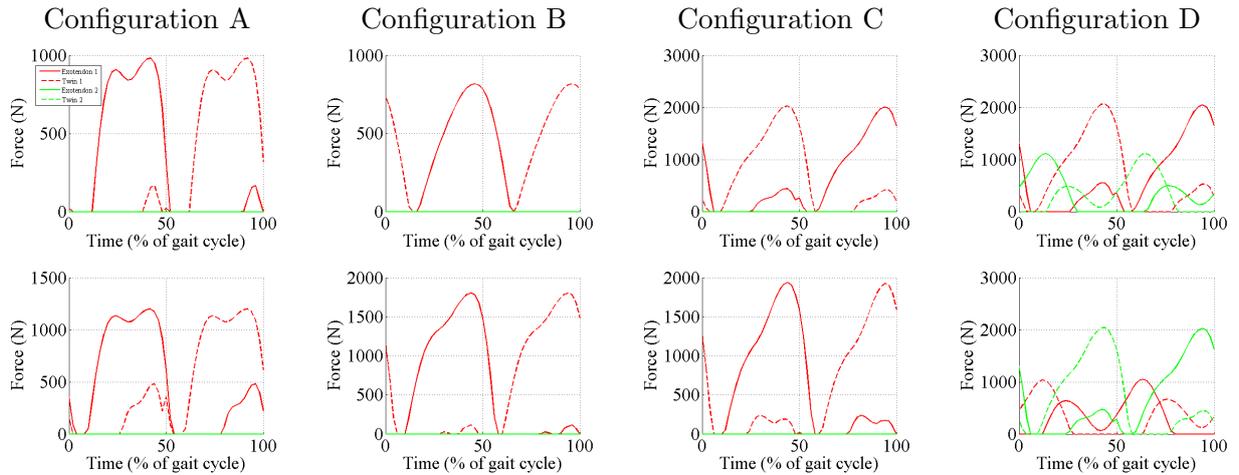
Forces, fast speed

Figure 35: Forces, first row: moment optimized, second row: power optimized. The twin exotendons are represented by a dashed line.

4.3 Global performance of each optimization

To show the influence of each of the 16 optimizations (four configurations for the three speeds and the global optimization), we can display the gain or loss of performance for the average residual moment and power with each of the optimizations.

A performance gain means that the moment/power provided by the exotendons reduce the moment/power without assistance at the joints. Conversely, a loss of performance gives rise to the opposite effect: the moment/power provided by the exotendons increase the moment/power at the joints without assistance and therefore, make the walking harder.

To do this, we represent, for the moment, for one given configuration:

$$100. \frac{\Delta M}{M} = 100. \frac{M - M_{initial}}{M_{initial}} \quad (9)$$

with $M_{initial}$ the average residual moment without contribution of the exotendons at a specific speed and M the average residual moment for a speed, with the design the best optimized for this configuration.

By analogy, for the power, the formula is:

$$100. \frac{\Delta P}{P} = 100. \frac{P - P_{initial}}{P_{initial}} \quad (10)$$

Thus, when these formulas are negative, there is a performance gain and when they are positive, there is a loss of performance.

Moment optimized

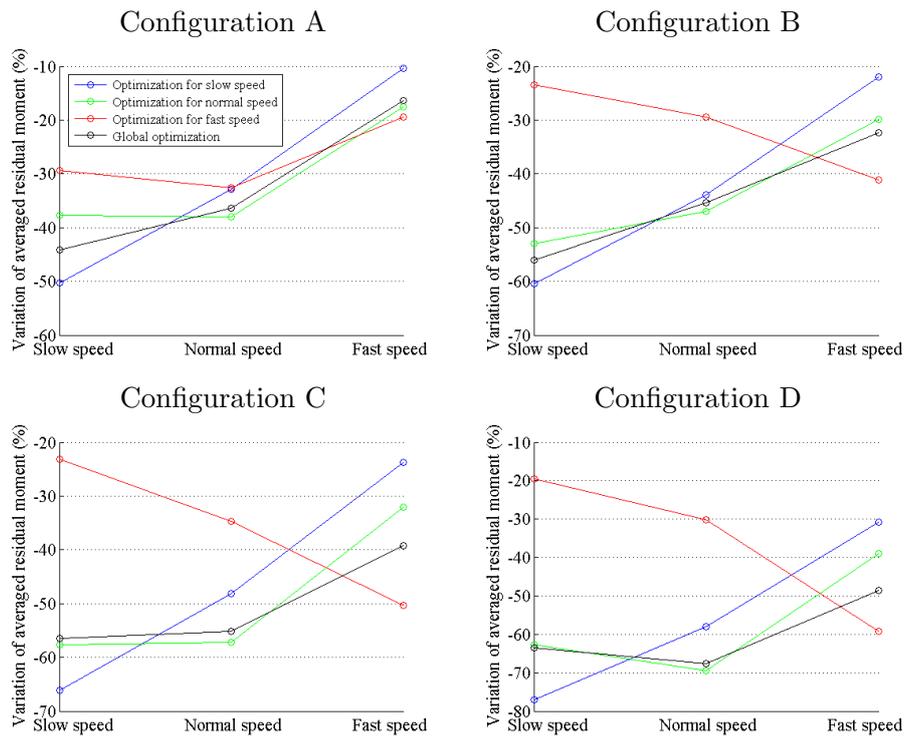


Figure 36: Performance of each optimized design for the moment.

Power optimized

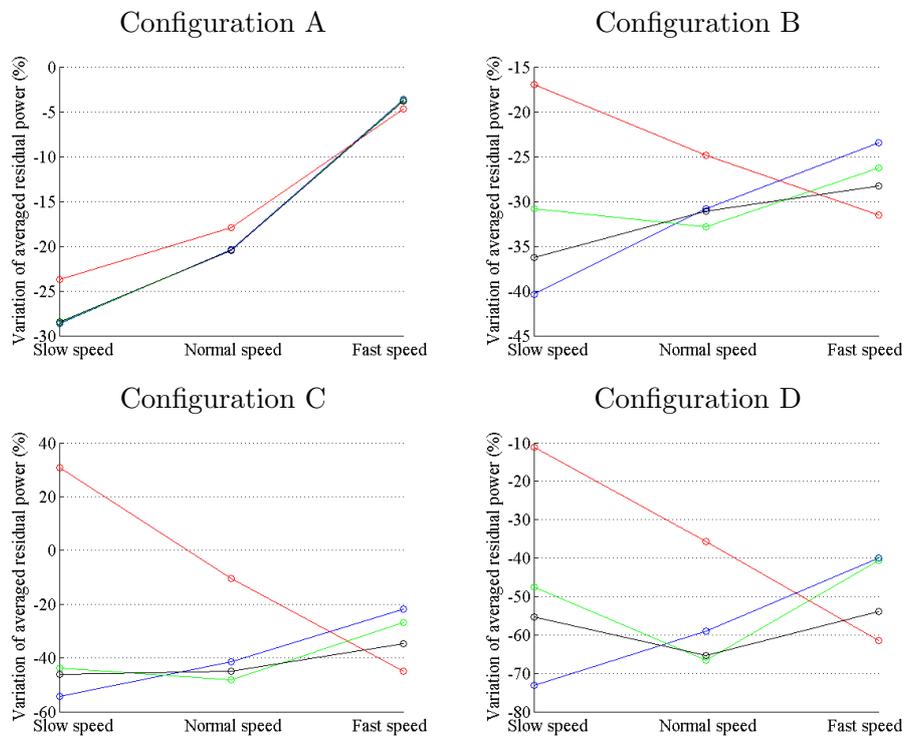


Figure 37: Performance of each optimized design for the power.

Interpretation

We can see that overall, the global optimization, i.e. for the three speeds is closer to the one optimized at the normal speed.

Also we can notice that the only optimization which can become counterproductive is the configuration C optimized for the fast speed, as it makes the walking harder at slow speed (power about 30% higher than the initial one). This is linked to the forces of the exotendons which are high (between 1000N and about 3400N) compared to the others because of the very small slack length (-21.3mm) (figure 15).

5 Simulation using Webots

A simulation with Webots was done in order to see the contribution of the exotendons in a more realistic situation. The objective of this part is thus to see if the exotendons have the expected effects, in a dynamical environment. This paves the way for more complex simulations.

5.1 Method

The positions and the moments of the joints which were used come from a subject, studied by the University of Twente.

The average residual moment without contribution of the exotendons is 23.51Nm.

The skeleton implemented in Webots has to follow the reference positions of the joints. To do this, moments are applied at the joints (feedback moments). They have to deal with the passive moment of the joints which come from the ligamentous system of the legs, and are supposed to be smaller when the moments provided by the exotendons are applied.

Then we can make an analogy with a control loop:

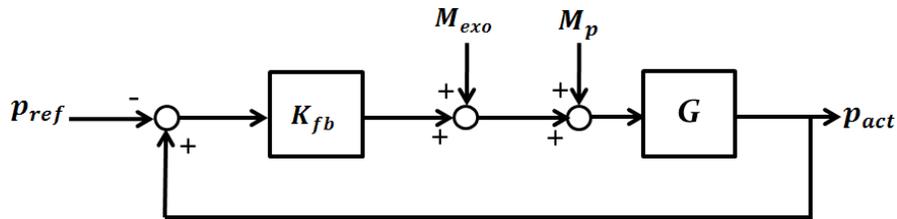


Figure 38: Control loop to follow the reference position of the joints.

with q_{ref} the reference positions of the joints which have to be followed, q_{act} the actual positions of the joints, K_{fb} the feedback moment gain, M_{exo} the active moment provided by the exotendons, M_p the passive moment (which actually also depends on the actual positions of the joints) and G the controlled system, i.e. the moment at the joints.

Thus, this moment is $M = M_{fb} + M_{exo} + M_p$, with $M_{fb} = K_{fb}(q_{act} - q_{ref})$ the feedback moment. To compute the active moment provided by the exotendons, first the optimization of the different configurations was done then it was implemented by using the same formulas as presented in the first section.

5.2 Optimization

Using PSO, the optimization was done for the average residual moment. The following results were obtained:

Leg	Right			Left			
Parameter [mm]	r_{ankle}	r_{knee}	r_{hip}	r_{ankle}	r_{knee}	r_{hip}	L_{slack}
A	-89.32	N/A	N/A	N/A	N/A	N/A	15.18
B	-51.35	2.89	27.55	N/A	N/A	N/A	0.13
C	0.90	-2.78	-8.39	-50.98	4.50	26.27	2.83
D exotendon 1	-2.25	-5.48	-6.25	-41.78	14.90	15.39	17.57
D exotendon 2	-50.63	-1.26	17.96	2.51	1.22	0.04	-0.22

Design	$C_{mom}(Nm)$
A	17.25
B	11.01
C	10.48
D	17.29

The optimization for the configurations A,B and C are consistant but for the configuration D the result is not interesting as it is higher than the the optimization of the configuration C.

The parameters of the designs A, B and C were implemented in Webots.

Without contribution of the exotendons, the position of the joints, the passive and the feedback moments are represented below:

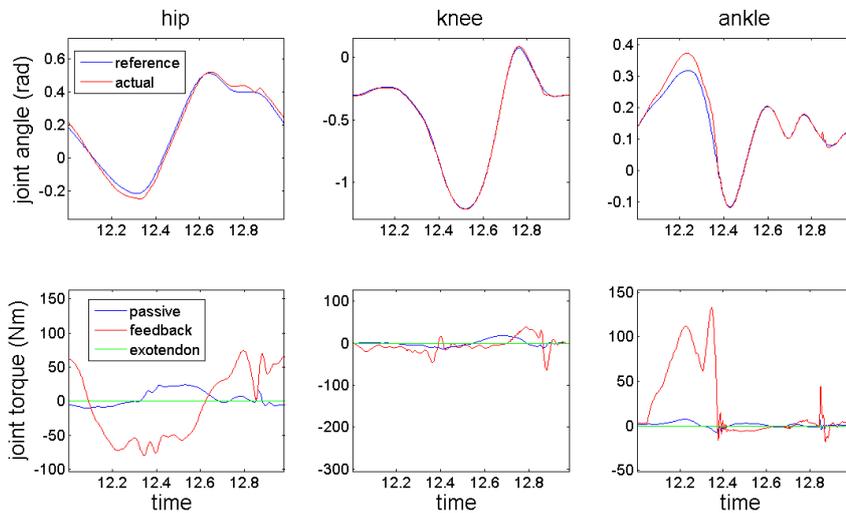


Figure 39: Position of the joints, passive and feedback moments on the right leg, without assistance, during one gait cycle.

These are the results with the exotendons:

Configuration A

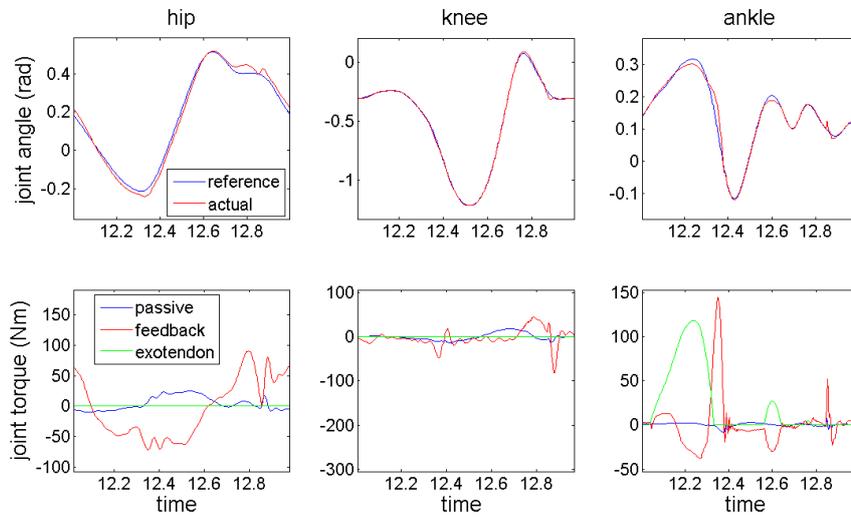


Figure 40: Position of the joints, passive, feedback and exotendon moments on the right leg, during one gait cycle.

We can see that the moment provided by the exotendon reduce the feedback moment (it works in the same way than the feedback moment without assistance), and that the reference position of the ankle is correctly followed.

Configuration B

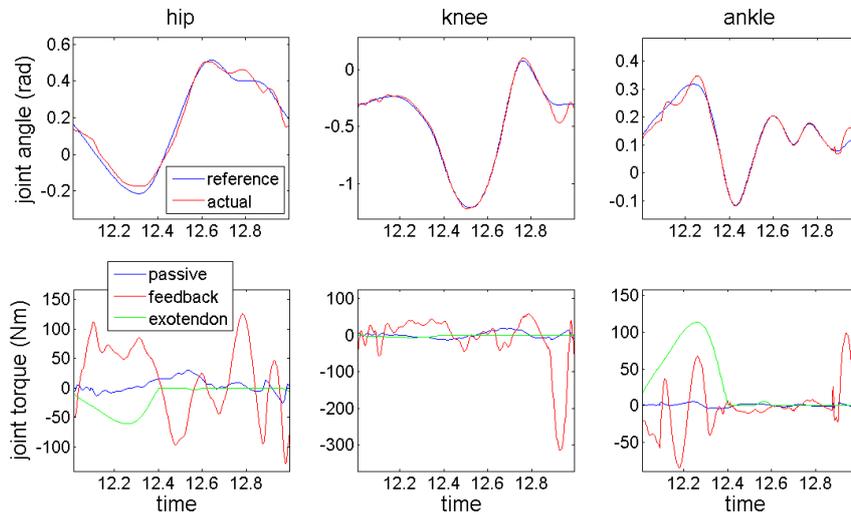


Figure 41: Position of the joints, passive, feedback and exotendon moments on the right leg, during one gait cycle.

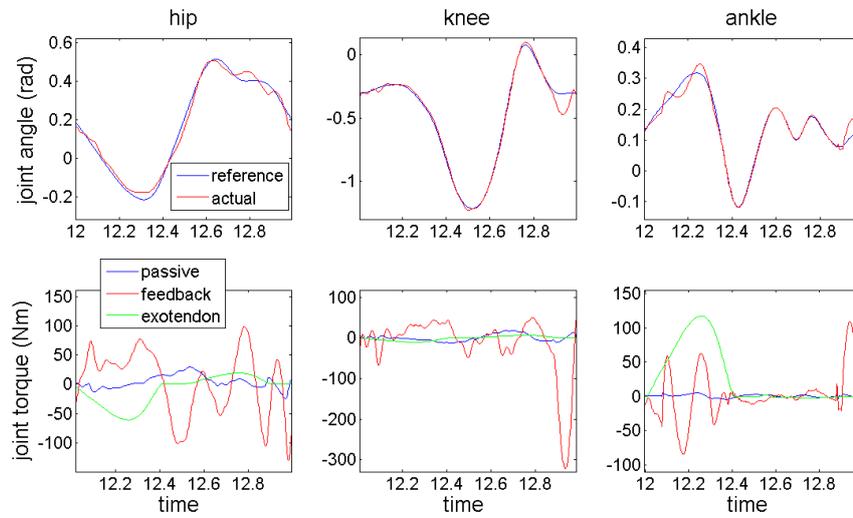
Configuration C

Figure 42: Position of the joints, passive, feedback and exotendon moments on the right leg, during one gait cycle.

For the configurations B and C, we can see that the moment provided by the exotendons goes in the same way than the feedback moment at the ankle without assistance but the ankle position is followed with difficulty as the feedback moment is fluctuant (in the range [12s;12.4s]). Moreover, as there is a peak of feedback moment at the end of the gait cycle, the actual position of the joint differs from its reference position.

For the knee there is nearly no assistance provided by the exotendons. Indeed, the radii of the pulleys attached to the knee are very small (array 5.2), so the moment also (equation 3). Moreover, we can observe the same peak of feedback moment at the end of the gait cycle which disturbs the following of the reference position of the joint.

For the hip the moment provided by the exotendons also goes rather in the same way than the feedback moment without assistance. However, compared to the case with no assistance, the feedback moment is still significant and work in the other way.

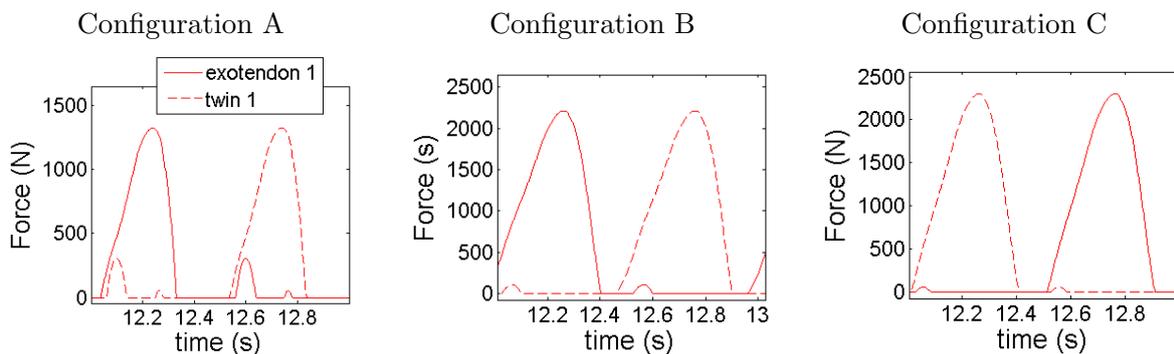


Figure 43: Exotendon forces

6 Conclusion

Upon completion of this project, I have fulfilled the specifications set by my supervisors, but the last part regarding the simulation with the Webots could be depth. Some results regarding the first part were different than the ones from the literature because Particle Swarm Optimization algorithm couldn't converge to the optimal optimization especially for the most complex designs.

During the project, my knowledge improved in the softwares Matlab and Latex and also I had the opportunity to discover the simulation software Webots.

It was interesting to implement an algorithm such as Particle Swarm Optimization and see the parameters which can influence the way the algorithm converges, and also the differences between this heuristic process with the other method, the Simulated Annealing.

For the dynamical simulation part, the implementation of the configuration A seems to give good result (position of the ankle correctly followed and good behavior of the feedback moment) and therefore would be probably the most applicable design to implement in the real world, as starting point.

The configurations B and C would be more difficult to implement. Indeed, the correct tracking of the reference positions of the joints is the most important criterion to validate such exoskeleton which does not have to prevent the gait cycle. Moreover, the feedback might not be comfortable to provide by the person.

7 Acknowledgments

I would like to thank my supervisors for their support and the help they gave me for the steps of the project as for the problems that I encountered for the using of the different softwares (Matlab, Latex, Webots, SVN...). Moreover, I am really grateful for the advice that they gave me to improve my report.

8 Further Work

The next step could be to machine a complete exoskeleton, probably to design the configuration A in order to try the exoskeleton in reality. Thus we could see if it really works as some points were neglected as the movements of the joints in the coronal and transverse planes.

9 Bibliography

References

- [1] http://en.wikipedia.org/wiki/Powered_exoskeleton.
- [2] Andrew Valiente: **Design of a Quasi-Passive Parallel Leg Exoskeleton to Augment Load Carrying for Walking**. Submitted to the Department of Mechanical Engineering in partial fulfillment of the requirements for the degree of Masters of Science at the Massachusetts Institute of Technology, August 2005.
- [3] Michael S. Cherry, Dave J. Choi Kevin J. Deng, Sridhar Kota, Daniel P. Ferris :**Design and fabrication of an elastic knee orthosis: preliminary results**. *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Philadelphia, USA*, September 10-13, 2006.
- [4] Antonie J van den Bogert: **Exotendons for assistance of human locomotion**. *BioMedical Engineering Online*, 2003.
- [5] Aaron M. Dollar, Hugh Herr: **Active Orthoses for the Lower-Limbs: Challenges and State of the Art**. *IEEE*, 2007.
- [6] Stansfield BW, Hillman SJ, Hazlewood ME, Lawson AM, Mann AM, Loudon IR and Robb JE: **Normalisation of gait data in children**. *Gait Posture*, 2003, 17:81-87.
- [7] http://en.wikipedia.org/wiki/Simulated_annealing.
- [8] Thomas Kämpfe:**Simulated annealing: Use of a new tool in bin packing**.*Annals of Operations Research*, 16 (1988) 327-332.
- [9] Premchand Akella: **Simulated Annealing**.
- [10] Dowsland: **Modern Heuristic Techniques for Combinatorial Problems**. *Simulated Annealing, Dowsland, Chapter 2 in Reeves*, 1993.
- [11] Marco A. Montes de Oca: **Particle Swarm Optimization, Introduction**. *IRIDIA-CoDE, Université Libre de Bruxelles (U.L.B.)*, May 7, 2007.
- [12] R. C. Eberhart and Y. Shi: **Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization**, 2000.
- [13] R. C. Eberhart and Y. Shi: **Computational intelligence and security, Part 1**. *International conference, CIS 2005, Xián, China*, December 2005.
- [14] D.A. Winter: **Biomechanics and Motor Control of Human Movement**. *Wiley, New Jersey, fourth edition*, 2009.

10 Appendix

Sampling periods by least squares approximation

We have to estimate Δ_{moy} in the following equation:

$$\Delta_i = \Delta_{moy} + \epsilon_i \quad (11)$$

where Δ_i represents the time separating the previous angular position of the joint at the time i and its next angular position and ϵ_i represents the error between Δ_i and Δ_{moy} .

Therefore, as we have 51 measures of the angle of the joints provided for the three gait cycles, we have to minimize the following quantity:

$$\chi^2(\Delta_{moy}) = \sum_{i=1}^{51} \left(\frac{\Delta_i - \Delta_{moy}}{\sigma_i} \right)^2 = \sum_{i=1}^{51} \omega_i (\Delta_i - \Delta_{moy})^2 \quad (12)$$

where $\omega_i = \frac{1}{\sigma_i^2}$ are the weighting factors of the measures Δ_i .

By differentiation, we can compute the minimum of $\chi^2(\Delta_{moy})$ which is:

$$\Delta_{moy} = \frac{\sum_{i=1}^{51} \omega_i \Delta_i}{\sum_{i=1}^{51} \omega_i} \quad (13)$$

With the weighting factors corresponding to the powers generated at the joints, we obtain the following formula:

$$\Delta_{moy} = \frac{\sum_{i=1}^{51} P_i \Delta_i}{\sum_{i=1}^{51} P_i} = \frac{\sum_{i=1}^{51} M_i \Omega_i \Delta_i}{\sum_{i=1}^{51} P_i} = \frac{\sum_{i=1}^{51} M_i \Delta \alpha}{\sum_{i=1}^{51} P_i} \quad (14)$$

where P_i is the power, M_i the moment and Ω_i the angular velocity at the joint at the time i . $\Delta \alpha = \alpha_{i+1} - \alpha_{i-1}$ is the angular difference between the next position and the previous one.