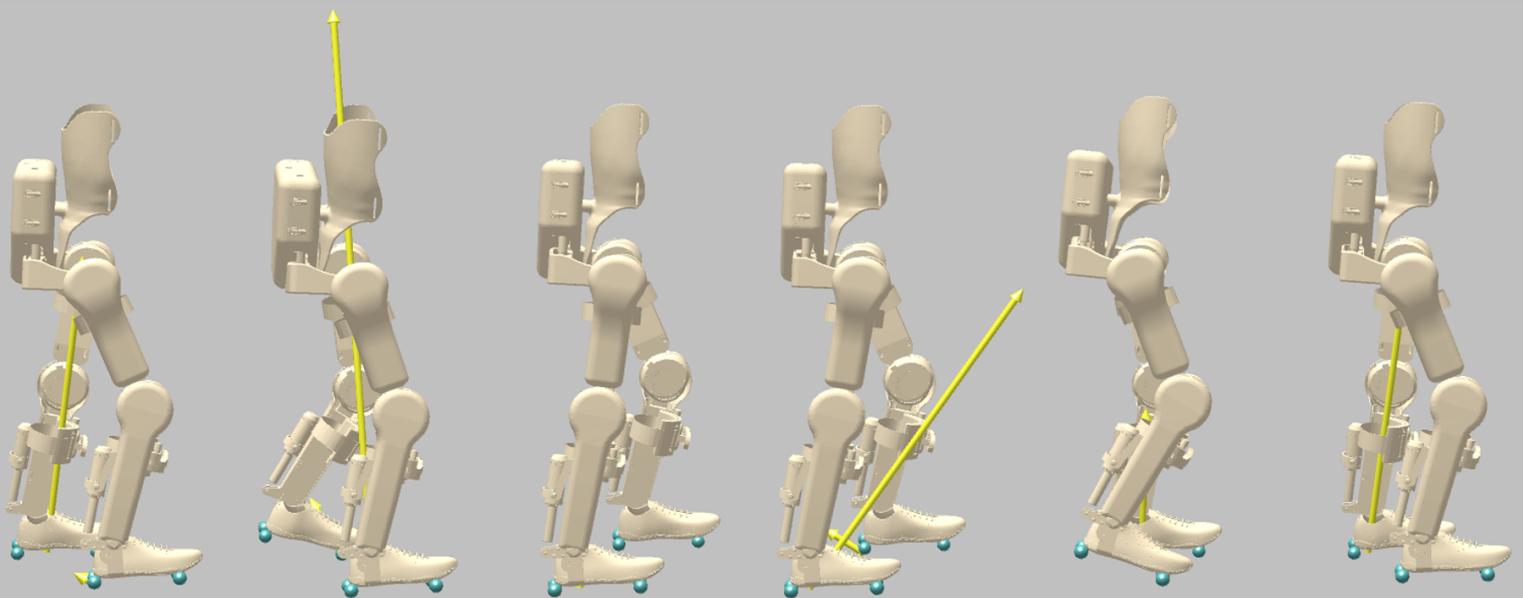


Towards dynamically stable gait of an 8 degrees of freedom exoskeleton through predictive forward dynamics simulation

U.N. Sheombarsing



Towards dynamically stable gait of an 8 degrees of freedom exoskeleton through predictive forward dynamics simulation

by

Umit Nishan Sheombarsing

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday April 2, 2020 at 14:30.

Student number: 4176030
Project duration: September 1, 2019 – April 2, 2020
Thesis committee: Prof.dr. F.C.T van der Helm, TU Delft, chair and supervisor
Dr. T. Geijtenbeek, TU Delft, daily supervisor
Dr.ir. R. Happee, TU Delft, external committee member

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

Preface

All good things eventually come to an end. And apparently, the word eventually is continuously open to interpretation. At least, it was for me. So here I am, after almost nine years I will be finally graduating. Here's a free tip for anyone who stumbles upon this report: finish your studies in 8 years! Combining graduation with a job on the side is tough and tiresome.

I want to start by expressing my gratitude towards my supervisors, Frans and Thomas, for guiding me through the past year of graduating. I want to thank Riender Happee as well, for completing my thesis committee as the external member.

The past year I was lucky enough to find support in a lot of friends and family. Foremost, I want to thank 'the squad' for having an infinite amount of coffee and lunch breaks, gamepies and woord-graptogrammen and providing me with useful feedback on my presentations. And...making sure I was there at 9.00 in the morning (which they gave up after a while).

I want to thank Thomas and Stijn for proofreading my complete thesis. Melina, Noor and Jorik for reading parts of my thesis and Tirza for helping me out with photoshop. I want to thank my roommates for being my complaint-walls and the rest of my friends for keeping my socially alive for the last year.

Last but not least, I want to thank my parents for providing me with the unconditional love and support for the past years. In the last year, more than one motivational gift was delivered at my house. The happy socks were able to cheer me up ('happy socks, happy thesis') as well as the occasional post card with slightly compulsory messages ('goals for 2020: find happiness, graduate, eat well, graduate'). I could not have done this without the both of you.

The research I conducted last year centers itself around exoskeleton devices. During the literature review I immersed myself in this field and conducted a review on 15 exoskeletons and identified their strong and weak points. I looked at the different ways that the exoskeletons are controlled, built and operated.

In the research project that followed, the goal was to create a simulation in which an exoskeleton model is walking autonomously. The process, results and findings of this research project are brought together into one article, which is the main content of this report.

*Umit Sheombarsing
Delft, March 2020*

CONTENTS

I	Introduction	1
I-A	State-of-the-art	1
I-B	Problem statement	2
I-C	Research goal	2
I-D	Approach	2
II	Model	3
II-A	Body parts and joints	3
II-B	Contact geometry	3
II-C	Adding human weight	3
III	Control algorithm	3
III-A	Straight locomotion	4
III-B	Perturbations	6
III-C	Standing balance	6
IV	Optimization	6
IV-A	CMA-ES	6
IV-B	Experiments	6
V	Results	7
V-A	Straight locomotion	7
V-B	Perturbed gait	9
V-C	Standing balance	10
VI	Discussion	11
VI-A	Locomotion stability	11
VI-B	Standing balance stability	11
VI-C	4 instead of 6 DoF per leg	11
VI-D	Torque requirements	12
VI-E	Model limitations and recommendations	12
VII	Conclusions	13
	References	13
VIII	Acknowledgements	14
	Appendix I: Optimized parameters	15
	Appendix II: Additional results on locomotion	16
	Appendix III: Standing balance position	17

Towards dynamically stable gait of an 8 degrees of freedom exoskeleton through predictive forward dynamics simulation

Umit Nishan Sheombarsing

Abstract—The main form of mobility for paraplegic patients is by wheelchair. However, not moving the legs comes with adverse health effects. Exoskeletons are one solution to get these patients walking again. One of the aims of exoskeleton research is the complete restoration of locomotion for paraplegic patients. The achieved gait must be stable, safe and comfortable for the patients.

Most research goes into exoskeleton devices which require the use of balancing aids. In the form of crutches, these aids help the exoskeleton users to maintain stability. One of the goals is to eliminate the reliance on balance aids and let the robot do most work. Until now only two exoskeletons are able to achieve autonomous dynamically stable gait. The gait generation algorithms used in these device are based on inverse dynamics. trajectories are calculated and closely tracked. The main challenges of inverse dynamics control algorithms are slow and static movement, balance recovery issues or computational complexity.

In this research the aim is to achieve autonomous walking without balance aids. The Project MARCH exoskeleton is taken as an example in this case study. This device has 4 actuated degrees of freedom per leg. The exoskeleton is modelled in OpenSim. Using predictive forward dynamic simulations, a gait algorithm is implemented and evaluated. The reflex-based control algorithm is based on proportional-derivative controllers. This control algorithm is implemented in SCONE and is optimized using the Covariance Matrix Adaptation - Evolution Strategy method. A second simulation experiment uses the same method to achieve standing balance.

After optimization of the control algorithm, dynamically stable gait patterns emerge. The exoskeleton model shows limit cycle behaviour and is able to walk for at least 30 seconds at a speed of 0.7 m/s. The controller can optimized to reject perturbations up to 300 N for 0.1 s. The emerging gait pattern shows two features, which complicate the implementation in the real exoskeleton. The model shows a back-heel rotation during stance phase and hits the joint limits during the liftoff phase. Standing balance is also achieved by a different controller.

This research serves as a proof of concept on using SCONE (or more general, predictive forward dynamic simulations) to simulate and test an autonomous exoskeleton. The algorithms are completely feedback controlled require no predefined trajectories. Certain features seen in the emerging gait patterns remain to be resolved. This work demands more research to prevent back-heel rotation, to avoid approaching the joint limits and model the toe-off more adequately in order to reduce the peak torque. Furthermore, interesting research can be done on a randomized perturbation rejection and on how to model the inelastic collision at the joint-ends properly as well as on making a comparison between the gait patterns presented in this research and the patterns currently used in the exoskeleton. Only if these challenges are addressed, the gait algorithm becomes eligible to employ in a real exoskeleton.

I. INTRODUCTION

Each year 250 new lower spinal cord lesions occur in the Netherlands, resulting in paraplegia. These patients are paralyzed from the pelvis to the feet and lose, among other things, the ability to stand and walk [1]. Losing these abilities can have great consequences on the patient's mental state. 70% of the patients report a reduced quality of life and satisfaction as consequence of mobility limitations [2]. The wheelchair is the main tool to improve the mobility of paraplegic patients. However, there are adverse physical consequences related to permanent wheelchair usage. These include a negative influence on metabolism and a higher risk on cardiovascular diseases [3]. One alternative to sitting in a wheelchair is walking through the use of robotic devices. These devices are called lower-limb exoskeletons. Patients using these kind of devices have reported improvements in spasticity, skin health, pain, diabetes, bladder and bowel function, and fat loss [4].

A. State-of-the-art

The research on lower-limb exoskeletons can be divided into two branches. The branch of training devices focuses purely on the aspect of movement and assumes devices are merely used in clinical environments. The robots have been designed to automate repetitive behaviour, alleviating the manual work of therapists during various stages of neurorehabilitation [5]. The second branch in this field (assistive devices) aims to completely restore abilities such as autonomous walking, complicating the task of movement with e.g. balance requirements, overcoming perturbations and safety precautions. Different literature studies have been conducted on exoskeleton control strategies [6]–[8]. Gardner et al. [9] provide an overview of the exoskeleton capabilities of the four current market leaders. They conclude that devices achieving higher gait speeds are more unstable and require the use of balance aids. The only device (considered in this research) that is statically stable, REX, shows a slow gait of 0.5 m/s, whereas humans walk at around 5 m/s. Another autonomous exoskeleton is made by ATALANTE. Both will be shortly described below.

1) **REX**: The REX exoskeleton by Rex Bionics was the first exoskeleton not requiring balance aids [11]. The gait is controlled using the Zero Moment Point (ZMP) concept, drawn schematically in figure 1.

An advantage of this method is the system being statically stable when not perturbed. By being permanently

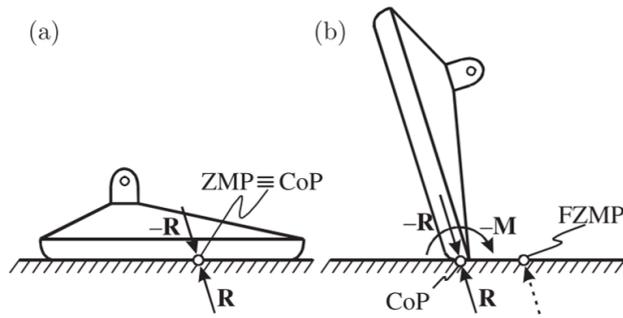


Fig. 1: The ZMP can be interpreted as at the point at which the net moment of the inertial and gravity forces are zero in the horizontal plane. (a) : In practice, this point coincides with the Center of Pressure (CoP) of the foot, as long as the ZMP lies within the span of the feet . The goal of ZMP control is to keep the ZMP within this area. (b) : When the CoP is at the foot's edge, the ZMP becomes fictitious (FZMP). The object is now unstable and will rotate around the edge to fall down[10]. This concept can be used in a feedback controller by regulating the CoP or in a more feed-forward manner creating reference trajectories that already ensure a safe positioning of the ZMP. Image from [10].

in a statically stable situation, this device doesn't fall down with power outages. The drawbacks of using ZMP control are slow gait speed and low perturbation rejection (the algorithm is meant to prevent balance loss, not recover from it) [12]. Therefore, the method has a high chance of falling when facing large perturbations due to the limited support area during single stance phase [13]. Opposed to human locomotion (whereas human gait is, roughly formulating, falling constantly and recovering balance [14]), the ZMP is a conservative control measure because of its requirement of perpetual static balance.

2) **ATALANTE:** The ATALANTE exoskeleton is another exoskeleton achieving autonomous locomotion [15]. Harib et al. [16] use the concept of hybrid zero dynamics and optimization to guarantee dynamically stable gait at 0.1 m/s. With this method, joint position trajectories are calculated and are closely tracked. The calculation of these trajectories suffer from high computational complexity. The research of Harib et al. shows the benefits of modern optimization techniques as they accelerate the calculation of joint position trajectories by a factor of 100 to 41 seconds. Even though this is a significant improvement, high computation times remain a challenge of this approach.

As stated by De Boer in [17] the hybrid zero dynamics paradigm relies heavily on complex offline optimizations to obtain a stable gait and only works if the robot can conform to the single prescribed gait. The hybrid zero dynamics controller has no way of explicitly handling rough terrain or large pushes that derive the exoskeleton from the pre-planned gait. Harib et al. address this issue by implementing an extra control layer, driven by supervised machine learning algorithms. The exoskeleton is able to overcome perturbations in the form of pushes of 30 Ns in

the sagittal plane during periodic gait.

While demonstrating the first ever realization of hands-free dynamically stable gait of an exoskeleton, the main challenges that remain are the high computation times and the requirement of predefined trajectories.

B. Problem statement

The main challenge in the exoskeleton research field is to overcome the need of balance aids. Thus far, two exoskeletons are able to achieve this: REX and ATALANTE. These device are controlled using algorithms relying heavily on prescribed trajectories. Stable joint position trajectories are calculated and used as main tracking reference. To achieve close tracking of these trajectories, the necessary joint forces, torques and motor inputs are computed. This is called inverse dynamics. A main challenge of these inverse dynamics methods is to overcome deviations from the trajectory.

C. Research goal

The aim is to obtain stable gait without balance aids for a simulated exoskeleton. This is achieved using predictive forward dynamics simulation.

D. Approach

In predictive forward dynamics simulations the opposite from inverse dynamics is done: a control model controls the motor inputs and the motion trajectories emerge from the control model. Predictive forward dynamics simulations have already been used to tune the reflexive control model in musculoskeletal models [18,19]. The control of such a model can be optimized which is done in [20]. The power of this method lies in the simplicity of the model: each actuator is assigned a certain function (governed by muscle models) and the emerging motion is analyzed and optimized. Adaptive locomotion emerges from feedback loops and is detached from trajectories or datasets.

In this research simulations are done of the Project MARCH exoskeleton. This device is designed by a group of students from the Delft University of Technology, with the aim to contribute to hands-free exoskeleton locomotion. The exoskeleton has 8 Degrees of Freedom (DoF). The modelling of the exoskeleton with human inside is done in OpenSim [21]. This software application enables users to model and simulate humans, exoskeleton or other assistive devices. An overview of the model in presented in chapter II.

Then this model can be used in SCONE, which is an open-source software application using the OpenSim toolbox to perform dynamic simulations. The SCONE platform also facilitates the implementation of custom-made control algorithms and optimization of model/control parameters. The model and control algorithm can be freely designed, while the optimization algorithm is implemented in the

software [22]. The control algorithm is designed by translating observations of locomotion patterns to control tasks and is presented in section III. The model parameters can be optimized according to different objectives such as gait stability and energy consumption. This is covered in section IV.

The concluding sections will present, analyze and discuss the results followed by a list of recommendations for further research.

II. MODEL

For the model of the exoskeleton and its dynamic simulations, the open-source toolbox OpenSim is used [21, 23]. This section will list the different features which are combined to create an exoskeleton model with human inside.

A. Body parts and joints

The exoskeleton consists of 9 segments and is drawn together with the axis convention in figure 2.

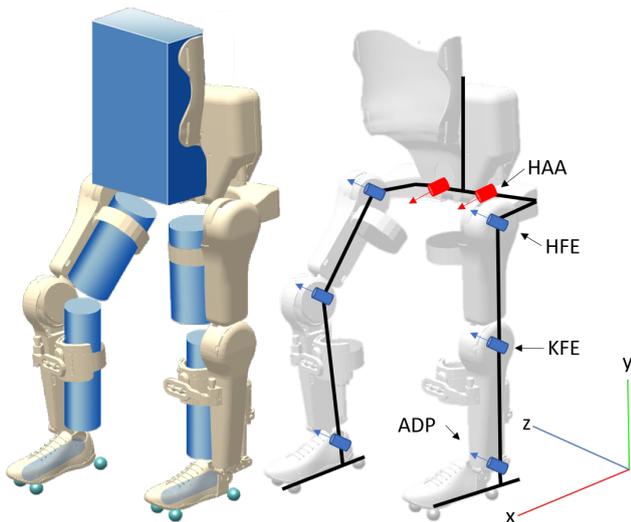


Fig. 2: This figure illustrates the MARCH exoskeleton with a human inside which is used for simulation. On the left side the complete model is drawn. The blue parts represent the body parts, rigidly linked to the exoskeleton. The turquoise dots below the foot represent the points at which contact between body and environment is simulated. The right half of the picture displays the set of body segments, linked by the joints. The hip ab-/adduction (HAA) joint rotates around the local x-axis, to move the hip in the lateral/frontal (yz-) plane. The blue joints rotate around the local z-axis to move the body in the sagittal (xy-) plane. HFE: hip flexor-extensor, KFE: knee flexor-extensor, ADP: ankle dorsiflexor-plantarflexor. There are no degrees of freedom in the transversal (xz-) plane.

The exoskeleton parts are connected by joints. Each joint is moved by an actuator. The range of motion (ROM) and maximum torque of each joint are listed in table I. The actuators can deliver both positive and negative torque. In the OpenSim model, the maximum torque is set to 500 Nm.

TABLE I: ROMs and torque limits. The axes of rotation are displayed in figure 2.

	Lower limit [deg]	Upper limit [deg]	Torque [Nm]
HAA	-17	17	150
HFE	-20	100	140
KFE	-115	5	140
ADP	-25	15	150

The joint limits are modelled as a spring damper system. In OpenSim these are named 'coordinate limit forces'. These limit forces are inactive when the joint is inside its range of motion. Through a transition region of 0.1 degrees, the limit forces and damping linearly increase towards a constant. The stiffness of the spring is 20 Nm/deg, the damping is set to 1 Nms/deg. In this study special care is taken not to harm the joint limits.

B. Contact geometry

In the model of figure 2, the ground is modelled as a half-space: the bottom half of the simulation environment acts as the ground. On the foot, four half-spheres are attached, which act as contact points for the feet. The impact between foot and ground is modelled by a Hunt-Crossley impact model [24]. This model is ready-made in OpenSim and the required parameters, which are comparable to asphalt [25], can be seen in table II

TABLE II: The parameters used to set up the ready-made impact model in OpenSim.

Parameter	Value
Stiffness	2e6 [N/m]
Dissipation coefficient	1 [-]
Static friction coefficient	0.9 [-]
Dynamic friction coefficient	0.8 [-]
Viscous friction coefficient	0.6 [-]

C. Adding human weight

In this study, the exoskeleton is loaded with extra weight as if the device is carrying a user. The user is modelled as separate solid bodies, rigidly attached to the device. In figure 2, these symmetric body parts are drawn.

III. CONTROL ALGORITHM

This section will elaborate on how the control algorithms are made for three experiments. The first two experiments focus on locomotion and perturbed locomotion. The last experiment is to achieve standing balance. A scheme displaying the hierarchy in the design of a controller is drawn in figure 3. The control algorithms in this research are designed to execute a desired behaviour pattern. Depending on the scenario (i.e. walking or standing) the motion is divided in phases. Each joint has a separate task in each phase. For each joint two types of tasks can be defined: reaching a global target (with respect to the world's inertial frame) or reaching a local target (which is expressed as an angle between two limbs).

The joints are controlled by proportional-derivative (PD) controllers. Each joint controller is defined according to

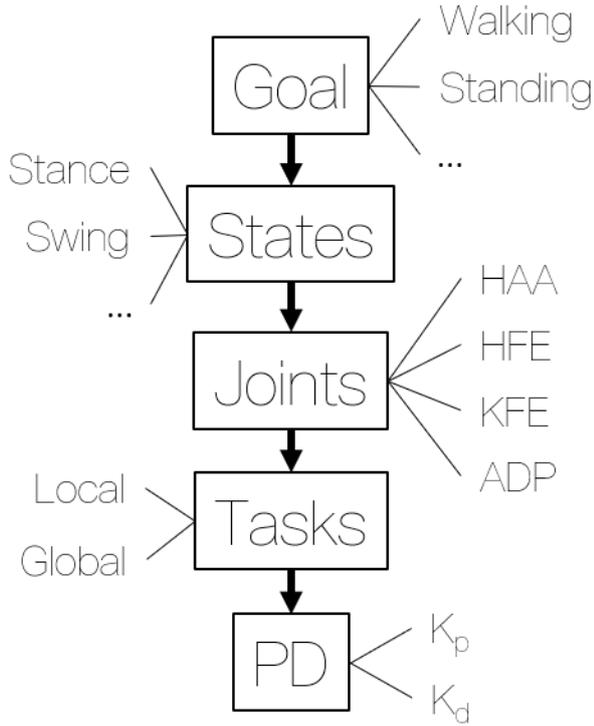


Fig. 3: The hierarchy in the design of a controller. From the desired behaviour pattern, the movement can be broken down in phases. In each phase, a joint can be assigned a task. These tasks are controlled using PD-controllers.

the control scheme of the PD-controller in figure 4. All gain parameters and goal parameters are estimated by the optimization method implemented in SCONE. Optimization will be done using the Covariance Matrix Adaptation - Evolution Strategy. To reach a desired behaviour pattern, optimization can be done on parameters in the lower four layers as drawn in figure 3. In this research, however, only reference targets for the PD controllers and gain parameters are optimized.

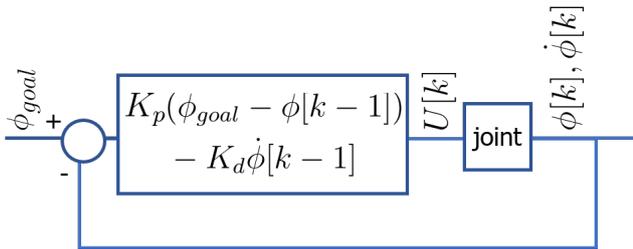


Fig. 4: A simplified PD control scheme, used to control the joints. The proportional gain K_p magnifies the controller input linearly with the difference between the goal ϕ_{goal} . The derivative gain K_d corrects this input according to the speed $\dot{\phi}$. The state at time step $k - 1$ is fed back to the control system.

A. Straight locomotion

The first step is identifying the gait phases. In this research, the gait cycle is divided in five phases for each leg, starting

from the heel strike, which marks the start of the early stance phase. An overview of these phases is drawn in figure 5.

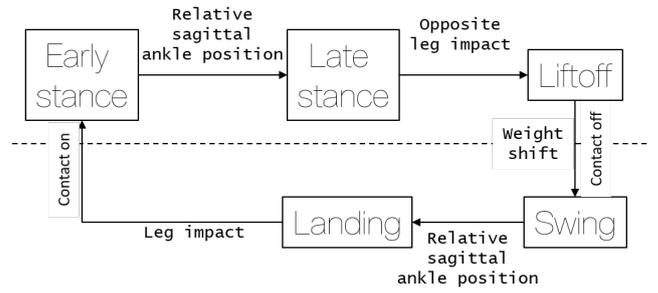


Fig. 5: The five phases in the gait controller and transition mechanisms. In the upper three states the foot has ground contact. The transition from early stance to late stance and from swing to landing are based on the relative position of the ankles, which can be optimized. The other three transitions initiate if a load threshold is reached, which can be optimized as well.

In the stance phases the leg fully supports the body. In the lifftoff phase, the leg must push the center of mass over the midstance and shift the weight towards the opposite leg. In the swing phase and the landing phase, the leg is moved forward while making sure the feet do not touch the ground and that the leg is placed at the correct position to retain balance. In the coming sections, each phase will be elaborated in order to determine a task for each joint during that phase.

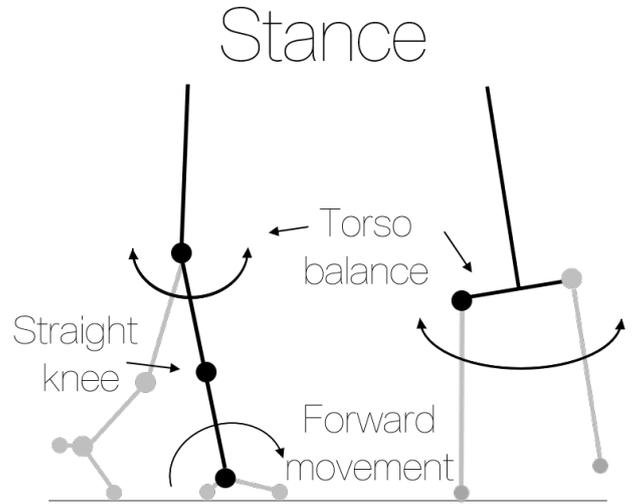


Fig. 6: In the early stance phase and late stance phase, the hip joint is maintaining torso balance in the two vertical planes. The knee joint must be kept nearly straight to prevent collapsing. The ankle actuator must prevent the joint from overstretching.

1) *Early stance and late stance, fig. 6:* In human gait, balancing the upper body is the main task of the hip muscles [26]. In this control algorithm, the task of balancing the upper body is assigned to the stance leg. The hip flexor and hip adductor are assigned a global control target for the torso. The target is to keep the torso balanced in both the sagittal

and frontal plane. The knee joint must prevent the leg from collapsing. A local control target is set for the knee actuator to keep a straight angle between the upper leg and lower leg. The ankle has virtually no involvement in maintaining balance during stance phase [26], but stabilizes the lower leg [27]. A local control goal is set to prevent the ankle from overstretching. The function of the stance leg is twofold: balancing the hip and prevent collapsing. According to [27], the net sum of all three torques in the sagittal plane must be positive to achieve the latter.

The early stance phase starts after the foot lands on the ground (and ground reaction force is greater than zero), during this phase the other leg is in swing or landing phase. The late stance phase starts when the stance leg is at right angle with the ground. In this research, the function of early and late stance is equal, but the late stance phase can also be used as double support phase.

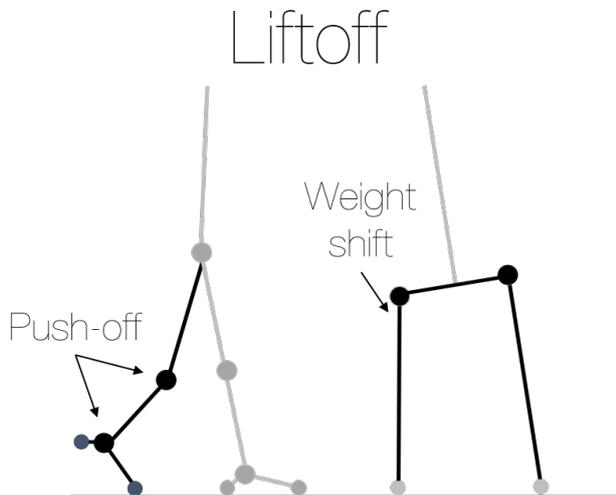


Fig. 7: During liftoff phase the leg must ensure enough forward velocity to overcome the equilibrium point. At the same time, weight must be shifted towards the other leg.

2) *Liftoff phase, fig. 7:* The liftoff phase is the phase where the ankle has to deliver torque to gain velocity in order to move forward [27]. At the same time, most of the body weight must be shifted to the opposite leg. The hip adductor is assigned a global control goal to shift the torso to the other side. The upper leg must be kept behind the body to direct the force in the right direction and is given a global target. The knee is given a local control goal to keep the knee slightly bent. The ankle joint must perform plantarflexion to extend and push the body forwards and is assigned a global target.

The liftoff phase starts when the opposite leg is starting early stance phase (i.e. ground reaction force is unequal to zero). By implementing a load threshold the liftoff phase can be initiated with a delay.

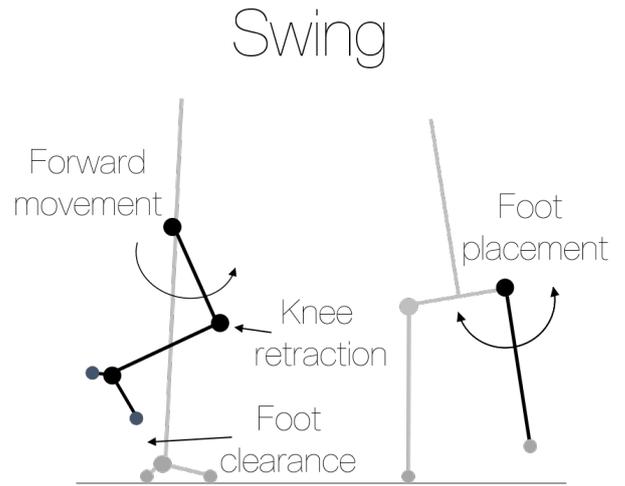


Fig. 8: While the other leg carries the complete body weight, the swinging leg must move forward while ensuring the foot does not strike the ground. In the lateral plane, the hip moves in an appropriate angle to stabilize.

3) *Swing phase, fig. 8:* From research on lateral balance control in human locomotion, it is known that humans must provide active control of lateral motion to maintain balance [28]. The hip abductor is assigned with a global control goal to recover dynamic balance. Inspired by the dynamical balance method described in [29], the joint target ϕ_{target} is adjusted proportionally with K_c to the lateral speed $V_{z,CoM}$ of the center of mass

$$\phi_{corrected} = \phi_{target} + K_c \cdot V_{z,CoM}. \quad (1)$$

The corrected goal ϕ_{target} is the value fed into the PD-controller. A comparable method of dynamic step-width adaptation has been implemented in an existing exoskeleton [30]. With the same formula ($V_{z,CoM}$ is swapped for $V_{x,CoM}$), a correction will be made on the swing leg target. The knee and ankle are assigned local targets to ensure sufficient ankle clearance. The swing phase starts when the complete body weight is carried by the stance leg.

4) *Landing phase, fig. 9:* The hip joints in the landing phase are assigned the same functionality as in swing phase. The knee and ankle are assigned global targets to place the lower leg and foot, respectively, in the right position for impact. The landing phase starts based on the relative sagittal ankle position, which can be optimized. In this case, the relative sagittal ankle distance is equal to zero: when the swing leg is passing the stance leg.

Overview

Each joint controller is governed by three parameters, except for the hip joints in swing phase where a joint target adjustment is made according to velocity. The algorithm is optimized to maintain a preset center of mass velocity, to prevent go beyond joint limits and to reduce torso tilt. To summarize, the five phases sum up to an

Landing

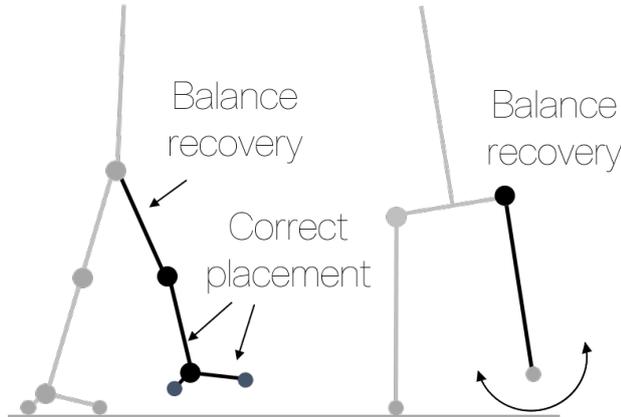


Fig. 9: Both hip flexion/extension and ab-/adduction are used to correct for balance disturbances and ensure a correct foot placement, while the rest of the leg must be brought in correct alignment for impact.

optimization problem where ((4 joints x 5 phases x 3 parameters) + 4 speed adaptation) = 64 parameters are optimized simultaneously. An overview of the complete set of parameters with the initialized values can be found in appendix I.

B. Perturbations

Perturbations are introduced in the simulation in a second experiment to test for robustness. Pulse forces incrementally increasing from 50 N that last 0.1 s are applied. These pushes are applied to the torso, in both forward and sideways direction. For each perturbation scenario an optimization is done to evaluate whether the model can return to its limit cycle.

C. Standing balance

During a third experiment, a controller will be devised to maintain standing balance for a given amount of time. The model will stand with its feet aligned.

The control algorithm is set up as follows:

- Each joint controls the limb above it using a PD controller. K_p and K_d are found through optimization.
- Initial global control goals are set to 0 degrees and found by optimization.
- Optimization objectives of the simulation are maintaining a vertical position, minimizing joint effort, sway distance and sway velocity.

To maintain unperturbed standing balance, research underlines the importance of the ankles [26]. It is observed that the net ankle moment can regulate the model center of mass. This control algorithm combines the same PD controllers as in the gait controller. This leads to an optimization problem where (4 joints x 3 parameters) =

12 parameters are optimized simultaneously. The initial parameters are tuned manually and can be found in appendix I.

IV. OPTIMIZATION

In SCONE different optimization scenarios can be created. Each scenario requires a model, as described in section II and a control algorithm, which is the feedback controller from section III.

In order to achieve desired behaviour (such as the optimization objectives mentioned in the previous section), we must design objective functions (i.e. loss- or penalty-functions). This objective function assesses how successful the chosen control/model parameters are in a scenario. An objective function is defined by a set of measures. A measure relates e.g. center of mass velocity or joint position to a penalty. The measures can individually be weighted to differentiate in importance during optimization.

Using predictive simulation it is possible to quickly iterate many parameter configurations to find an optimal configuration for a specific task. SCONE optimizes a scenario, using the Covariance Matrix Adaptation - Evolution Strategy (CMA-ES) algorithm [31], a nonlinear optimization algorithm. This section will elaborate on the CMA-ES method and how the optimization scenarios are set up in this research.

A. CMA-ES

The parameters that undergo optimization are initialized by a initial value and a search region, represented by a mean value and a search region. The search region is defined as the covariance matrix around the mean. The initial parameters are tuned manually. A sample of parameters is taken from the search region and the objective function is evaluated for those parameters. For each parameter set, the fitness is known. The optimization algorithm uses an evolution strategy: the algorithm changes the search region in the direction of the best performing parameters. After a number of iterations (depending on the problem complexity), a sufficiently working parameter set is found, after which the search region decreases gradually. The optimization is stopped at the point where no further progress is made in minimizing the cost function. A schematic representation of this process can be found in figure 10.

B. Experiments

This part will list the objective functions used in the three experiments conducted in the research.

1) *Locomotion:* In the locomotion scenarios only the gait controller from section III-A is optimized. The initial position is a position from which it is possible to converge to gait pattern, rather than starting from a standing position. The control parameters are initialized by manual tuning each phase (see appendix I). The objective function consists of three parts, described in detail in table III. The gait measure returns a value from 1 to 0, indicating, respectively, that that the model falls over immediately or that the required

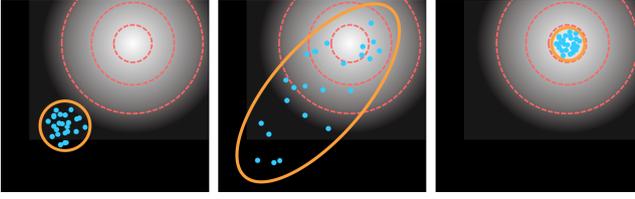


Fig. 10: A simplified representation of CMA-ES optimization of one parameter. The lighter areas mark a higher fitness of the objective function. In the left image the parameter is initialized with a mean value and a search region. The middle image shows an adaptation of the search region towards the optimal value. The right image shows a decrease of the search region, around the optimal value to eventually find the appropriate parameter. Image adapted from [32].

velocity is met. This measure is the predominant objective to achieve locomotion. Other objectives are restrictions on joint limits (hip ab-/adduction and ankle dorsi-/plantarflexion) and on torso tilt during gait. These measures holds for both locomotion experiments performed in this research.

TABLE III: The gait measure is the predominant factor in the objective function. HAA and ADP measures impose a penalty on surpassing the joint limits of the hip abductors and the ankle joint, respectively. The torso tilt measure restricts the torso from leaning too far backwards or forwards. The latter three measures impose a squared penalty with error E .

	Weight	Max	Min
Gait measure	100	-	0.7 m/s
HAA	1	17°	-17°
ADP	1	25°	-15°
Torso tilt	0.1	3 °	-10 °

2) *Standing balance:* The initial position of the standing balance controller has the legs aligned and all body parts pointed straight up. The main balance measure in this experiment is the balance measure. The balance measure checks whether the body center of mass is above a specified height. The remaining measures are set in place to prevent the joint velocity and body sway; which contribute to the goal of holding the body as still as possible. An overview the measures in this scenario is presented in table IV.

TABLE IV: The balance measure is the predominant factor in the objective function. ADP(-force) and HAA penalize exceeding the joint limits. Torso roll and tilt ensure a straight, slightly forward bent torso orientation. HFE-velocity and KFE-velocity reduce the angular velocity in the hip flexor and knee joint, respectively. The sway measure sums the distance travelled by all body parts and must be minimized as well.

	Weight	Max	Min
Balance measure	100	-	0.8 m
ADP	10	25°	-15°
ADP-force	10	0.01 N	-0.01 N
HAA	1	0.001°	-0.001°
Torso roll	0.01	-1°	-10°
Torso tilt	0.01	0°	0°
HFE-velocity	0.01	0.01°/s	-0.01°/s
KFE-velocity	0.01	0.01°/s	-0.01°/s
Sway	0.01	-	-

V. RESULTS

In this section, the results of three experiments will be presented. The first experiment is a 30 second simulation of the model walking. In the second experiment, perturbations are applied to this model. The final experiment shows the result of the model obtaining standing balance.

A. Straight locomotion

In the first experiment the model is walking undisturbed in a (nearly) straight line for 30 seconds. The model walks 20 meters without falling, tripping or deviating severely from its nominal step pattern, as drawn in figure 11. A compilation of moment captures during the simulation can be found in figure 13. The stable, cyclic behaviour of each joint is further illustrated by the phase portraits in figure 14. The optimized control values can be found in appendix I.

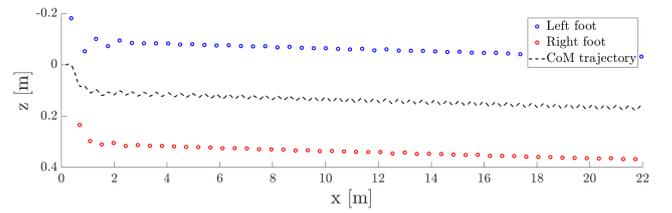


Fig. 11: The projection of foot placement and the center of mass trajectory over a duration of 30 s. After stabilizing from the perturbed initial situation, the model walks in a slightly diagonal direction, but in a straight line with a forwards speed of 0.7 m/s.

1) Phase analysis:

a) *Early stance and late stance:* When the foot strikes the ground, the lower leg is in a forward tilted position. From that instance, the three sagittal joints deliver the sufficient torque to prevent collapsing. The net torque of these joints is drawn in figure 12. During stance phase the hip joints have to ensure stability of the torso, drawn in figure 15.

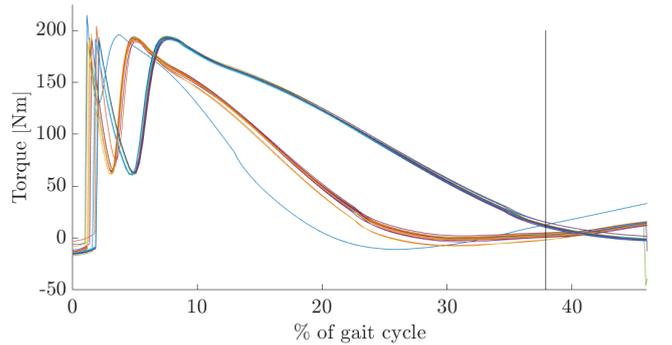


Fig. 12: The summed net joint torque of the three joints in the sagittal plane during stance, each line represents one step in the 30s simulation. The positive net torque prevents the stance leg from collapsing [27]. The vertical line at 38% represents the start of late stance phase.

The point at which the ground reaction force vector applies to the foot is called the center of pressure (CoP). Figure 16 draws the trajectory of the CoP during stance phase.

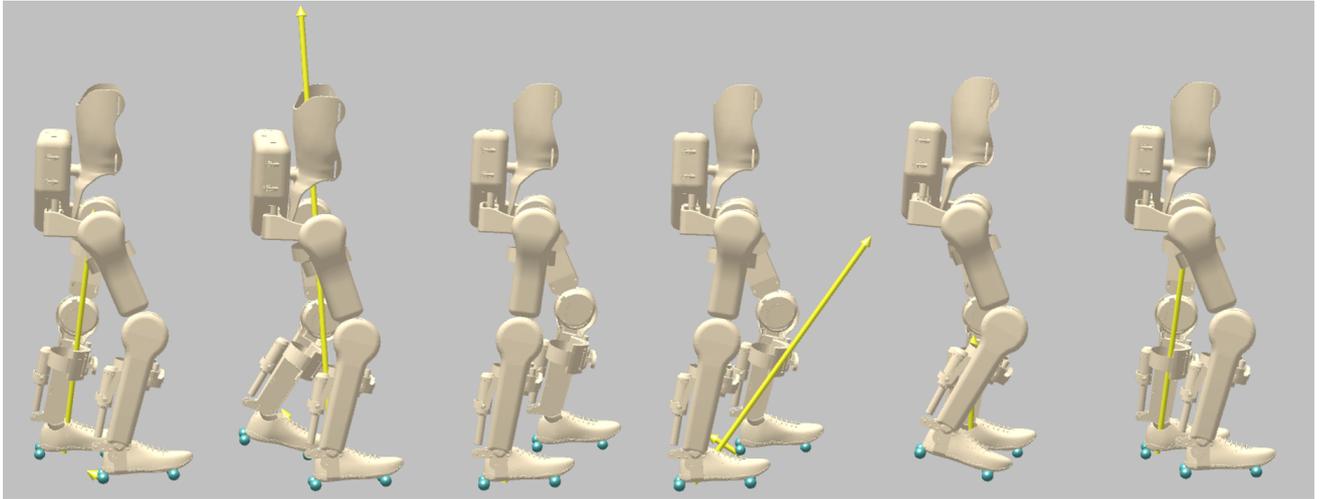


Fig. 13: This sequence of images shows a complete gait cycle of the right leg. From left to right: foot strike, early stance, late stance, liftoff, swing, landing.

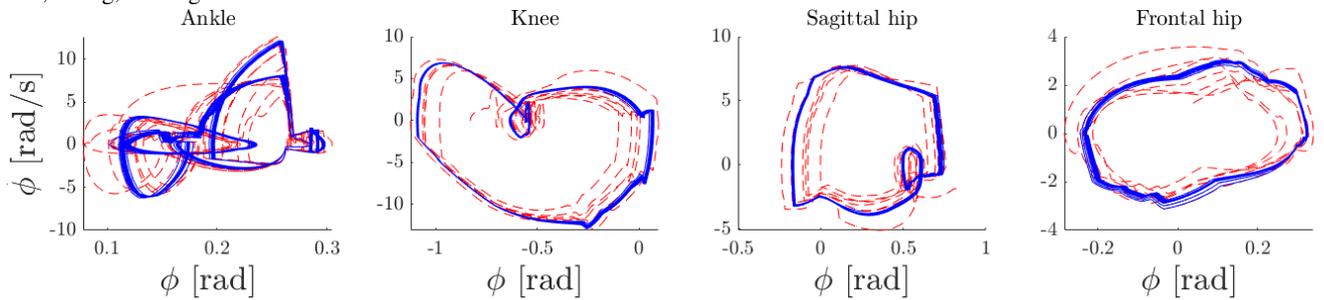


Fig. 14: The phase portraits of each actuated degree of freedom are drawn, during a simulation of 30 seconds. The model moves at a speed of 0.7 m/s. The red lines indicate the start from initial position after which the joints show limit cycle behaviour, in blue.

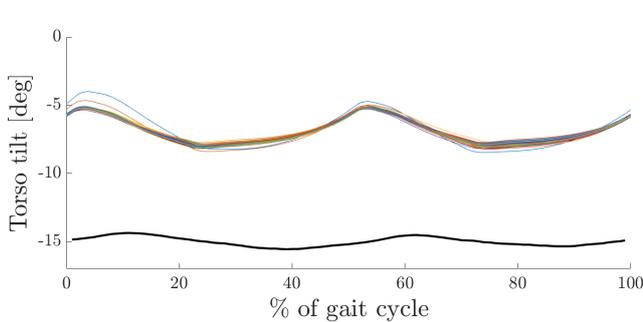


Fig. 15: The torso tilt for each step, oriented forwards throughout the complete gait cycle, each colored line represents one step in the 30s simulation. The black line draws human behaviour, as measured in [33]. Compared to human behaviour, the exoskeleton model shows a greater range of motion and an offset of 10 degrees.

At point 3 in figure 16 the foot rotates around the back of the heel. This movement is possible due to a rapid extension of the knee, thereby reducing the ground reaction force (the model is almost mimicking a jump). During this short period of 'semi-jumping', the friction force is overcome to rotate the foot. The rotating behaviour is further illustrated in figure 17. The rotation is always in the outward direction of the stance leg and shows symmetric behaviour. The model retains a fairly straight path.

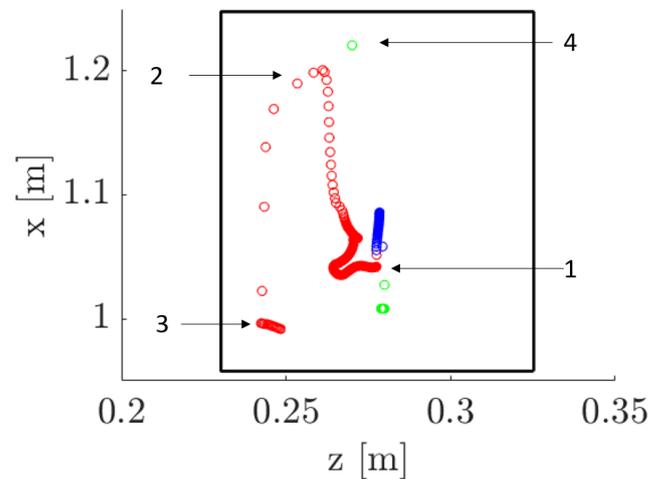


Fig. 16: The right foot (represented approximately by the black square) and the center of pressure trajectory during one step. The red circles represent early stance phase, followed by blue (late stance) and red (liftoff). At point 1 the heel strikes the ground and the CoP moves through a semi-circular in the middle of the foot towards point 2. At point 3, moves towards the back-heel, where the foot rotates around the heel. In late stance phase, the CoP moves back towards the middle, before moving to the toe when the foot is lifted off ground at point 4.

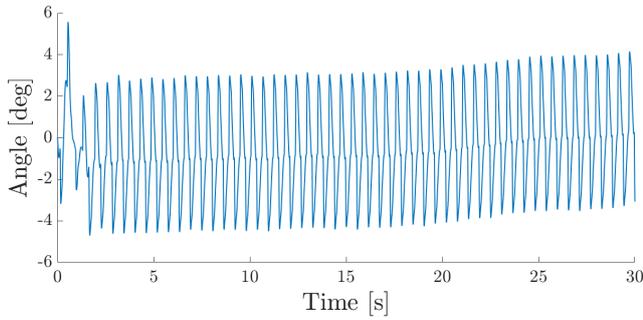


Fig. 17: The global orientation of the exoskeleton model during a 30 second simulation. The model stabilizes after 2 seconds. From then, the model rotates at every step. The resulting average direction changes very little.

b) Lifftoff phase: The hip ab-/adductor joints shift the weight towards the other side. This results in a stronger hip roll (or hip obliquity) of the upper body, compared to human locomotion.

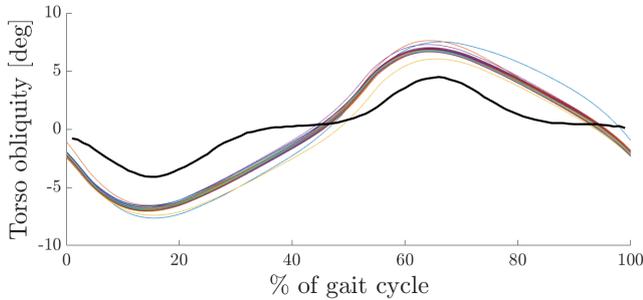


Fig. 18: The torso roll in simulation compared to human data in black from [33]. The model in simulation shows a greater torso obliquity, mostly to shift the weight to the opposite leg, which is entering stance phase.

In the lifftoff phase, there is a short moment of a forward ground reaction force of around 0.2 s (see the fourth frame of figure 13). There is no distinct toe-off, which is further illustrated by the ankle joint trajectory in figure 20. In this particular gait, the ankle hits its joint angle limit and starts lifting off afterwards, without showing any plantarflexing behaviour.

c) Swing phase and landing phase: The model starts in a perturbed situation (i.e. a configuration that does not exist in the nominal gait cycle). By adjusting step length and step width, the model eventually comes to a regular pace after four steps, see figure 11. The step sizes after stabilization are dependent on the center of mass velocity. The average step length is 22.9 ± 6 cm, the average step width is 39.9 ± 3 cm.

During swing phase, the knee and ankle must ensure sufficient foot clearance to avoid tripping. Figure 19 draws the foot clearance for each step during the simulation.

In landing phase, the target goal of the lower leg is -12.4 deg, indicating a forward tilted position and resulting in the leg striking the ground with bent knees, this can be seen in the right leg of the first frame in figure 13.

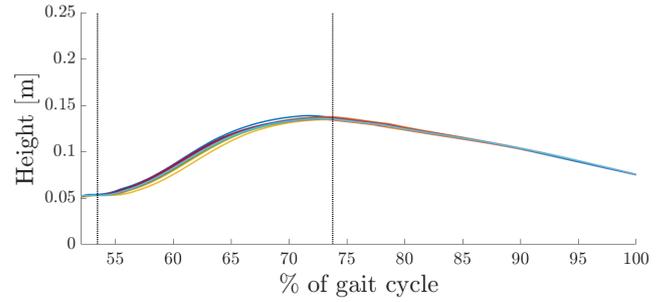


Fig. 19: This graph draws the height of the foot center of mass from swing phase (indicated by the vertical line at 55 %), through landing phase (second vertical line), until heel strike. The combined effort of the three sagittal joints results in sufficient foot clearance to prevent tripping. The graph doesn't reach 0 m, since these plots draw the CoM and not the contact point height.

2) Joint trajectory analysis: In figure 20, the joint trajectories of the four joints are displayed and compared to human behaviour. The knee and hip joint don't violate the joint limits of the exoskeleton. The knees do not stretch as far as humans do, resulting in bent knees during landing. While the knee and hip joints are qualitatively in agreement with human locomotion (except in landing phase), the ankle finds itself in a continuously dorsiflexed position. During lifftoff, the ankle violates the exoskeleton range of motion and reaches a dorsiflexion by 3 degrees (the model uses the passive coordinate limit forces to prevent joints from overstretching).

3) Torque analysis: During initial landing, the foot requires a peak torque of 160 Nm to prevent collapsing. During lifftoff, the knee requires a peak torque of 400 Nm. For this particular gait the exoskeleton actuators do not deliver the required torque.

B. Perturbed gait

In this experiments perturbances of varying magnitude are applied in both sagittal direction and lateral direction. The perturbations are applied at the torso. It is expected that the model alters its foot placement, to recover from the push within a few steps. The goal is to test the robustness of the controller algorithm, by gradually increasing the perturbation force. All pushes are applied for 0.1 second. Figure 22 shows an example, where the robot is pushed sideways twice during a 20 second simulation.

The model could resist the perturbation with a maximum force of 300 N. This holds for both types of perturbances. The model recovers within a few steps. The phase plots of these simulations can be found in appendix II.

When the push force in the sagittal plane was increased beyond 300 N, the model did not have enough time to place the foot far enough and falls down immediately. Increasing the push force in the sideways direction resulted in a model that was able to stay upright a few steps before turning unstable. These results are shown in appendix II.

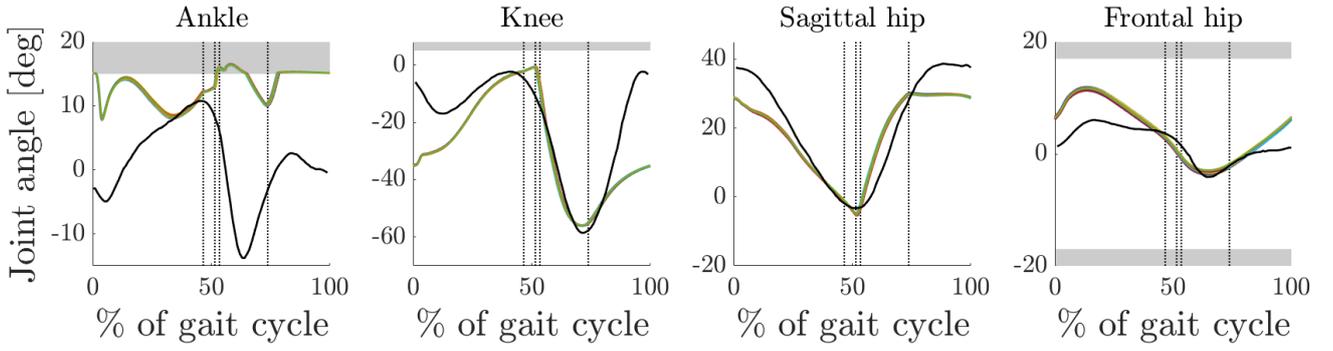


Fig. 20: Joint trajectories for each step, compared to human data in black from [33]. At 0 %, the leg starts the early stance phase. The vertical lines indicate the phase transitions. The gray areas represent the joint positions beyond the physical joint limits. The ankle is in gray area and uses the coordinate limit forces as joint end stop. The knee and hip joints stay well within the allowed range of motion. The knee does not stretch before landing phase, resulting in bent knee gait.

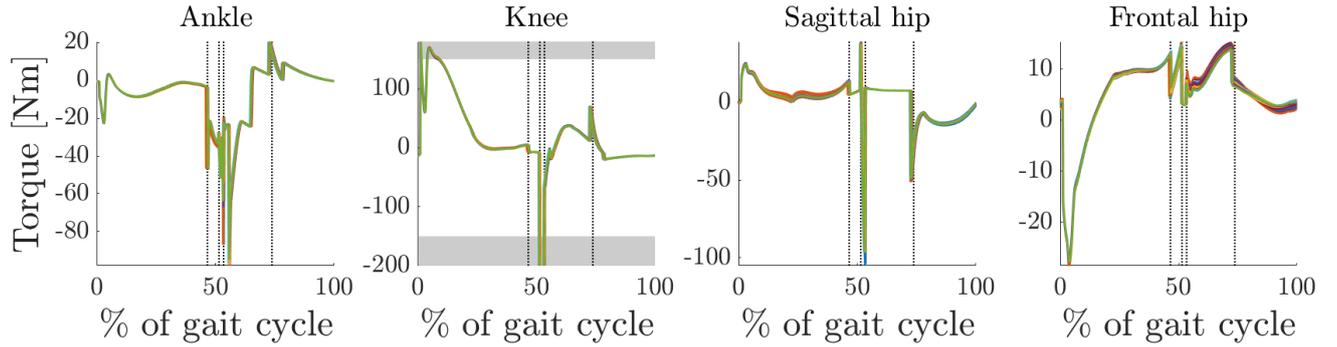


Fig. 21: The torque patterns required for the presented gait pattern. In the ankle and hip joints, torque patterns never exceeds 150 Nm. The knee however requires a peak torque at impact (0 %) and during liftoff phase.

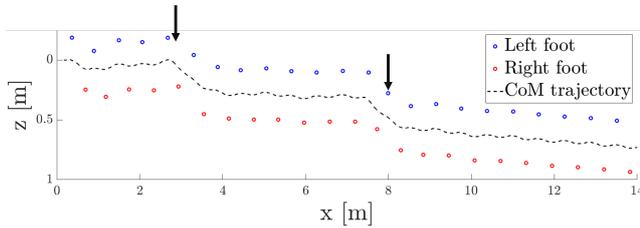


Fig. 22: Model foot placement while undergoing perturbations. The black arrows indicate the instances at which a push of $300 \text{ N} \times 0.1 \text{ s} = 30 \text{ Ns}$ is applied to the pelvis. The first perturbation is applied in mid-stance of the right foot. The left foot corrects for the push by decreasing step width, after which the right leg increases step width to recover completely. The second push is given in mid-stance of the left foot. The right leg reacts immediately on the changing CoM velocity by placing the leg more outwards, after which nominal gait is continued.

C. Standing balance

TABLE V: The target goals of each of the limbs, indicating a bent-knee equilibrium configuration. (z) indicates a rotation in the sagittal plane, (x) indicates a rotation in the lateral plane. LL = Lower Leg, UL = Upper Leg.

	LL (z) [deg]	UL (z) [deg]	Hip (z) [deg]	Torso (x) [deg]
ϕ	-3.79	0.69	-9.34	0.00

Standing balance is achieved in simulation by keeping the center of mass above the feet. The complete body is kept

upright by slightly bending forward the lower legs and the torso. The exact angular positions are displayed in table IV. A snapshot of the standing balance posture can be seen in appendix III. Figure 23 shows the sagittal distance of the center of mass and center of pressure in 20 second quiet standing experiment as exhibited by human behaviour.

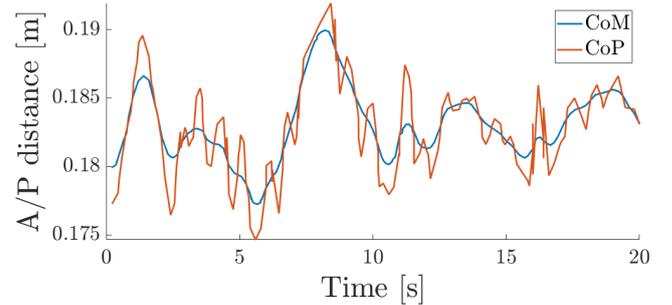


Fig. 23: Center of mass and center of pressure trajectories in human standing balance, measured by a pressure plate. The center of pressure oscillates around the center of mass to keep the CoM balanced. Data from [34].

Figure 24 shows the CoM and CoP trajectories in a perturbed standing balance experiment. A forward push of 120 N was applied to the torso for 0.1 second. Even though the model is now oscillating, it doesn't fall over. In trial experiments with greater pushes, the control algorithm is not able to withstand the perturbation.

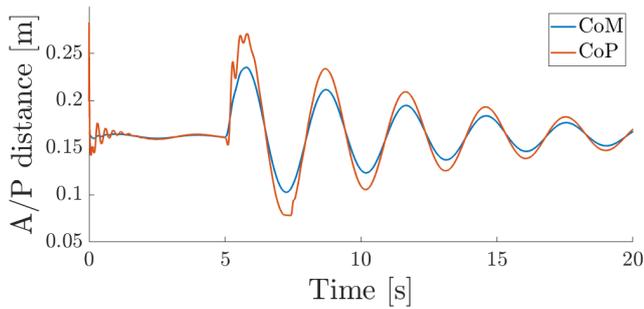


Fig. 24: After the controllers reach their target positions, the model gets perturbed by a 12 Ns force, applied to the torso in forward direction. This happens at the mark of 7 seconds. The model is able to withstand the push, but keep oscillating the center of gravity between the heel and foot.

In figure 25, the joint torque trajectories are drawn. The majority of correction is done by the ankle and the knee. The torque trajectories are in phase, resulting in oscillatory motion of the model, rather than a quick return to the target position.

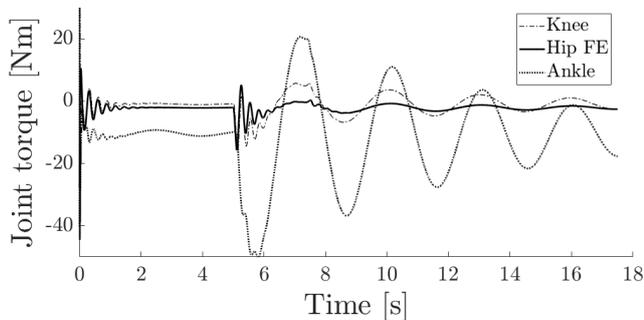


Fig. 25: The torque patterns of the three sagittal plane joints to maintain balance without perturbation and with perturbation (after $t = 5$). The required joint torque never exceeds 150 Nm.

VI. DISCUSSION

The goal of this research was to achieve stable gait and standing balance for an 8 DoF exoskeleton model. In this section the controller stability and controller behaviour will be evaluated according to the results presented in the previous section. Subsequently, an analysis will be done on the amount of DoFs and the torque requirements of the movement patterns. This section will be concluded by listing the limitations of this research and the resulting recommendations.

A. Locomotion stability

In the current state-of-the-art of exoskeletons, most control methods are based on generating stable joint position trajectories and carefully tracking them. These trajectories are tracked by the exoskeletons. By using forward dynamics controllers we omit the use of trajectories. Limit cycle behaviour is obtained shortly after initiating the simulation from a perturbed situation and maintained after undergoing an impulse force perturbation.

In this study, active sagittal and lateral control of the hip ab/adductors regulate the foot placement based on the center of mass velocity. After applying small perturbations, the model will return to its limit cycle behaviour within two steps. This robustness emerges from the control algorithm, in which foot placement is partially determined by the center of mass velocity. This research also puts focus on rejecting lateral perturbances, using a method comparable to [30] for a continuous adaptation of the step-width. Lateral step-width adaptation is not found in the exoskeletons that are able autonomous locomotion [11, 15].

In the case of the ATALANTE exoskeleton, sagittal perturbations are rejected by an extra control layer, based on supervised machine learning. This controller aims to return to the predefined trajectories both quickly and safely. When comparing the perturbation rejection abilities in this research to the supervised machine learning controller in [15], it must be stated that the machine learning algorithm outperforms the perturbation rejection presented in this research as they are able to reject force impulses of 750 N. However, the perturbation rejection presented in this research is inherent to the control model, rather than provided by an additional control layer on top of the nominal control.

In this experiment a scenario was created for each perturbation. Each of these scenarios was optimized independently. A common error in learning algorithms is overfitting: a model learns the to deal with the exact scenario and perturbation rather than being robust to perturbations in general. A possible improvement to this experiment is to create one scenario with multiple, randomized perturbations. The perturbations must vary in magnitude, timing and direction. By doing this, a more robust control algorithm can be obtained.

B. Standing balance stability

In the third experiment, standing balance of the model is achieved. The model undergoes small perturbations and is able to pertain balance. However, the model does not come to a standstill with the current controller, but keeps oscillating. This is a direct result of the in-phase torque trajectories. More research must be done on how to quickly return to the original position.

In this experiment, only one perturbation magnitude is applied at a regular interval. A possible improvement to improve robustness is to apply perturbations of randomized magnitude and with randomized interval. A second set of perturbations must be used to validate the algorithm.

C. 4 instead of 6 DoF per leg

As compared to the human, the exoskeleton model does not enable hip rotation in the transversal plane nor ankle eversion in the frontal plane. The most striking difference between human locomotion and the results presented in this research is the instance at which both feet are in contact with the ground, the double support phase. During this phase, the weight of the body must be shifted towards the front leg. In

human locomotion, this motion is governed by frontal plane motion of the ankle and hip as well as transversal plane motion of the hip (i.e. hip endo-/exorotation). The lack of two of these degrees of freedom, shows as follows

- The weight shift is done by increasing torso obliquity (or torso roll) in the frontal plane. As drawn in figure 18, the torso shows a range of motion twice the size compared to human locomotion.
- In human gait, the ankle is using its plantarflexor muscles to increase the angle between shin and foot. By doing this just before the foot lifts off ground, the body is pushed diagonally towards the front leg. The diagonal motion can not be performed in this model. This results in a reduced range of step lengths. In other words: the diagonal motion increases the range of step length. A smaller step length also requires the bent-knee landing which is seen in the locomotion pattern. The lack of toe-off also prevents the achieving higher speeds, since dynamic walking models can walk faster by pushing off harder, resulting in larger steps [35]. The step length in this model lies around 23 cm, which is one third of the average human step size [36].

A second consequence of the lack of ankle in-/eversion is during outward/inward foot placement. The foot lands on either side. Since there is no ankle to align the foot, the model has to 'roll' over the foot's edge, to recover full foot contact. This is, however, something that can be possibly overcome in the real prototype as the users wears sneaker boots. The shoes are not made of rigid material and allow for deforming and rolling over to the side.

D. Torque requirements

To achieve the gait pattern in this research, the peak torque required can not be generated by the prototype actuators. As a first step to solve this, the gait algorithm can be optimized for energy in an attempt to spread out the required torque over a longer amount of time. A second solution might, again, be a better modelling of the liftoff phase (to diminish the torque peak at the start of this phase).

E. Model limitations and recommendations

This section will focus primarily on the limitations of this research specifically.

1) *What is not modelled?:* The model used to do the simulations in this research, is a simplified representation of the real-world problem. Among other things, simplifications were made regarding

- Friction: the joints are assumed to rotate frictionless. Friction models that describe the friction between rotating bodies can be used [37] to model the real-world joints and actuators of which the properties are known (or can be measured). Introducing friction will result in higher torques to perform the same motion.
- Measurement uncertainties: the algorithm assumes perfect measurements from the actuators and joints. Measurement uncertainties can be taking the controller input from a binomial distribution around the actual

joint value. Introducing these uncertainties might improve the robustness of the control algorithm.

- Elastic deformation of the exoskeleton: every component is assumed to show completely rigid behaviour in the simulations. These elastic deformations can be modelled by very stiff joints/springs at the most flexible device parts. Introducing extra degrees of freedom will result in greater instability of the model. This could be resolved by enforcing stricter measures on the optimization.
- User-robot interaction: in this model, the user is represented by loose masses, attached rigidly to the exoskeleton parts. By doing this, the interaction between exoskeleton and user is completely neglected. The joint range of motion of paraplegic patients can be different compared to a healthy person. In this research, no model on human impairment is implemented. Only the exoskeleton limitations were taken into account.
- User preferences: the presented algorithm is merely focused on achieving dynamically stable gait. No preferences from a potential exoskeleton user regarding comfort, perceived safety are taken into account. Possible factors of discomfort might include the exaggerated torso tilt and roll. These preferences can be added as measures to the optimization problem.

2) *Joint end stops:* Each joint is limited in its range of motion by so called 'coordinate limit forces'. If the joint angle lies within the range of motion, this force does nothing. When going beyond the boundary, the coordinate limit force acts as a spring damper system. The stored force moves the body back in the right direction. These forces are not taken into account in the analysis of the joint torque. In this study, one of the optimization targets is to prevent going beyond the joint range of motion. As can be seen during lift-off in figure 20, the ankle range of motion is not sufficient and the joint limit is reached. The model makes use of these forces in its gait cycle. This does not implicate per se, that the range of motion of the exoskeleton joints is too small in general. It is, however, too small for the resulting gait in this study, in which a distinct toe-off is not performed.

Trial scenarios were made to minimize the coordinate limit forces. However, this assumes the joint limits do not 'exist'. More research must be done on modelling the inelastic collisions at the joint limits. Reaching the joint limit might not be preferable for the actuator, too hard collisions might inflict damage on the exoskeleton. An option to prevent this collision is, for example, placing a spring at the joint limits (to mimic the coordinate limit force), with an increase range of motion of the ankle.

A direct consequence of this 'free' energy storage, facilitated by the limit forces, is that this particular model is not ready yet to be optimized with an objective that minimizes total energy. In trial experiments, objectives were put in place to minimize the cost of transfer (i.e. energy used over distance). The result of this was, however, a large dependence on these limit forces with the ankle being beyond its limit at all times, which is behaviour that cannot arise in

the real-world prototype. An energy criterion is counteracting the joint limits.

3) *Uncontrolled behaviour / heel spin*: As elaborated in figure 16 and figure 17, the model rotates around the heel during stance phase, even though the exoskeleton itself does not allow for any rotation in the transversal plane. This behaviour might implicate that stability is found by mimicking a non-existent degree of freedom, which suggests the need for a hip endo-/exorotation to increase stability. It might also be a result of the center of mass being behind the foot's edge. The model is toppling backwards, but has enough forward velocity to overcome the backwards imbalance. Whether either of both causes is the case or not, this behaviour is uncontrollable and should be prevented. Further research must be done on how to prevent this.

4) *Optimization of model parameters*: In this study, only optimization was done on the parameters used in the control algorithm. However, SCONE itself is not limited to fitting just these control parameters. Model parameters (e.g. limit forces or the parameters indicating a phase transition) can be optimized as well. More intricate objective functions have to be set in place to, to improve stability and to prevent showing unfeasible gait patterns.

5) *Comparison with existing gait patterns*: The MARCH exoskeleton is currently controlled using predetermined trajectories. These trajectories are designed around the reliance of crutches to maintain balance. A comparison regarding joint position trajectories and energy consumption might provide useful insights on the benefits of this research.

VII. CONCLUSIONS

The aim of this research was to use predictive forward dynamics simulations to achieve stable behaviour for an 8 degrees of freedom exoskeleton. To that end comprehensible building blocks are combined in SCONE resulting in complex, nonlinear behaviour. SCONE combines the OpenSim platform parameter optimization and customized control algorithms to successfully control an exoskeleton. The OpenSim platform provides an insightful way of building up the exoskeleton model based on an existing prototype. It also facilitates the physics engine, allowing biomechanical simulation. On the other hand, SCONE has a ready-made optimization algorithm that relieves the researcher from implementing the algorithm themselves. The researcher is only left with devising the appropriate algorithms and defining optimization criteria.

In the control algorithm the gait cycle was subdivided in five phases. In each of these phases, control goals were set for the four separate joints. The control goals were initialized using basic knowledge of locomotion. The joint controllers were set up using basic knowledge of control systems, resulting in the extensive use of PD controllers. The optimization criteria are set up according to straight-forward rules and foremostly based on observations.

The elegance of this method lies in combining apparently trivial elements to achieve complex behaviour. Through this method stable gait is achieved, as well as stable standing

balance. An analysis was done on walking with 4 DoF per leg. The liftoff phase 'suffers' the most from the lack of hip rotation and ankle eversion. Another consequence is the step length of the model. Model limitations were identified as well as topics for further improvements. The next steps can involve modelling joint end stops, preventing uncontrolled behaviour and optimizing other parameters than just the control parameters.

This research can serve as a proof of concept on the validity of the method. This method allows for deeper research on:

- *Exoskeleton configuration*: by simulating an exoskeleton model, it can serve as a benchmark for realizing a working prototype. A preliminary analysis can be done on e.g. the required actuator power or joint range of motion. The model can also be gradually increased in complexity and fidelity by introducing extra DoFs, in the form of passive or active elements. These elements can be optimized as well.
- *Control algorithms*: in this study a joint is assigned one control goal at a time, regulated by one PD controller. This can be extended to more intricate control algorithms.
- *Different scenarios*: in this research only gait and standing balanced were achieved, but by altering the OpenSim environment and creating new control algorithms to perform more complex tasks, research might be done on e.g. uneven ground walking, stair climbing or sideways walking.

With this report set in place, more insight can be gained in to autonomous locomotion for exoskeletons and at the end of the line, hopefully, a wide employment of the robotic devices to restore mobility functionality for paraplegic patients.

REFERENCES

- [1] J. H. B. Nijendijk, M. W. M. Post, and F. W. A. Van Asbeck, "Epidemiology of traumatic spinal cord injuries in the Netherlands in 2010," *Spinal Cord*, vol. 52, no. 4, pp. 258–263, 2014.
- [2] H. I. Krebs, N. Hogan, M. L. Aisen, and B. T. Volpe, "Robot-Aided Neurorehabilitation," *Transactions on rehabilitation engineering*, vol. 6, no. 1, pp. 75–87, 1998.
- [3] "Zitten is het nieuwe roken - Zilveren Kruis," <https://www.zilverenkruis.nl/zakelijk/blog/mentaal-sterk/paginas/zitten-is-het-nieuw> accessed on 2020-03-16.
- [4] P. R. Geigle and M. Kallins, "Exoskeleton-Assisted Walking for People With Spinal Cord Injury," *Archives of Physical Medicine and Rehabilitation*, vol. 98, no. 7, pp. 1493–1495, 2017.
- [5] A. Esquenazi and M. Talaty, "Robotics for Lower Limb Rehabilitation," *Physical Medicine and Rehabilitation Clinics of North America*, 2019.
- [6] A. Esquenazi, M. Talaty, and A. Jayaraman, "Powered Exoskeletons for Walking Assistance in Persons with Central Nervous System Injuries: A Narrative Review," *PM and R*, vol. 9, no. 1, pp. 46–62, 2017.
- [7] T. Yan, M. Cempini, C. M. Oddo, and N. Vitiello, "Review of assistive strategies in powered lower-limb orthoses and exoskeletons," *Robotics and Autonomous Systems*, vol. 64, pp. 120–136, 2015.
- [8] M. R. Tucker, H. Bleuler, O. Lamercy, A. Pagel, R. Gassert, J. Olivier, V. Heike, R. Reiner, M. Bouri, and d. R. M. Jose, "Control Strategies for Active Lower Extremity Prosthetics and Orthotics : A Review," *Journal of NeuroEngineering and Rehabilitation*, no. January 2015, 2015.

- [9] A. D. Gardner, J. Potgieter, and F. K. Noble, "A review of commercially available exoskeletons' capabilities," *2017 24th International Conference on Mechatronics and Machine Vision in Practice, M2VIP 2017*, vol. 2017-Decem, pp. 1–5, 2017.
- [10] M. Vukobratovic and B. Borovac, "Zero-Moment Point — Thirty Five Years of Its Life," *International Journal of Humanoid Robotics*, vol. 01, no. 01, pp. 157–173, 2004.
- [11] G. Barbareschi, R. Richards, M. Thornton, T. Carlson, and C. Holloway, "Statically vs dynamically balanced gait: Analysis of a robotic exoskeleton compared with a human," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, vol. 2015-Novem, pp. 6728–6731, 2015.
- [12] H. Dallali, *Modelling and Dynamic Stabilisation of a Compliant Humanoid Robot, CoMan*. PhD thesis, University of Manchester, 2012.
- [13] L. Li, K. H. Hoon, A. Tow, P. H. Lim, and K. H. Low, "Design and control of robotic exoskeleton with balance stabilizer mechanism," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2015-Decem, pp. 3817–3823, 2015.
- [14] D. A. Winter, A. E. Patla, and J. S. Frank, "Assessment of balance control in humans," *Medical progress through technology*, vol. 16, no. 1-2, pp. 31–51, 1990.
- [15] O. Harib, A. Hereid, A. Agrawal, T. Gurriet, S. Finet, G. Bo  ris, A. Duburcq, M. E. Mungai, M. Masselin, A. D. Ames, K. Sreenath, and J. W. Grizzle, "Feedback Control of an Exoskeleton for Paraplegics: Toward Robustly Stable, Hands-Free Dynamic Walking," *IEEE Control Systems*, vol. 38, no. 6, pp. 61–87, 2018.
- [16] A. Agrawal, O. Harib, A. Hereid, S. Finet, M. Masselin, L. Praly, A. D. Ames, K. Sreenath, and J. W. Grizzle, "First Steps Towards Translating HZD Control of Bipedal Robots to Decentralized Control of Exoskeletons," *IEEE Access*, vol. 5, 2017.
- [17] T. De Boer, *Foot Placement in Bipedal Locomotion*. 1 ed., 2012.
- [18] H. Geyer and H. Herr, "A Muscle-Reflex Model that Encodes Principles of Legged Mechanics Produces Human Walking Dynamics and Muscle Activities," *Ieee Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 1, pp. 263–273, 2010.
- [19] S. Song and H. Geyer, "Generalization of a muscle-reflex control model to 3D walking," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, no. Eec 0540865, pp. 7463–7466, 2013.
- [20] T. Geijtenbeek, M. van de Panne, and A. F. van der Stappen, "Flexible muscle-based locomotion for bipedal creatures," *ACM Transactions on Graphics*, vol. 32, no. 6, pp. 1–11, 2013.
- [21] S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, and D. G. Thelen, "OpenSim: Open-source software to create and analyze dynamic simulations of movement," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 11, pp. 1940–1950, 2007.
- [22] T. Geijtenbeek, "SCONE: Open Source Software for Predictive Simulation of Biological Motion," *Journal of Open Source Software*, vol. 4, no. 38, p. 1421, 2019.
- [23] et al. Seth A, Hicks JL, Uchida TK, Habib A, Dembia CL, Dunne JJ, "OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement.," *PLoS Computational Biology*, vol. 14, no. 7, 2018.
- [24] K. H. Hunt and F. R. Crossley, "Coefficient of Restitution Interpreted As Damping in Vibroimpact.," *American Society of Mechanical Engineers (Paper)*, no. 75 -APM-H, pp. 440–445, 1975.
- [25] "Friction and Friction Coefficients," https://www.engineeringtoolbox.com/friction-coefficients-d_778.html, accessed on 2020-03-20.
- [26] D. A. Winter, "Human balance and posture control during standing and walking," *Gait & Posture*, vol. 3, pp. 193–214, 1995.
- [27] D. A. Winter, "Overall principle of lower limb support during stance phase of gait," *Journal of Biomechanics*, vol. 13, no. 11, pp. 923–927, 1980.
- [28] C. E. Bauby and A. D. Kuo, "Active control of lateral balance in human walking," *Journal of Biomechanics*, vol. 33, no. 11, pp. 1433–1440, 2000.
- [29] A. L. Hof, "The 'extrapolated center of mass' concept suggests a simple control of balance in walking," *Human Movement Science*, vol. 27, no. 1, pp. 112–125, 2008.
- [30] S. Wang, L. Wang, C. Meijneke, E. Van Asseldonk, T. Hoellinger, G. Cheron, Y. Ivanenko, V. La Scaleia, F. Sylos-Labini, M. Molinari, F. Tamburella, I. Pisotta, F. Thorsteinnsson, M. Ilzkovitz, J. Gancet, Y. Nevatia, R. Hauffe, F. Zanow, and H. Van Der Kooij, "Design and Control of the MINDWALKER Exoskeleton," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 2, pp. 277–286, 2015.
- [31] N. Hansen, "The CMA Evolution Strategy: A Comparing Review," *Towards a New Evolutionary Computation*, vol. 102, no. 2006, pp. 75–102, 2007.
- [32] "CMA-ES - Wikipedia," <https://en.wikipedia.org/wiki/CMA-ES>, accessed on 2020-03-16.
- [33] M. Kadaba, H. Ramakrishnan, and M. Wootten, "Measurement of lower extremity kinematics during level walking," *Journal of orthopaedic research*, vol. 8, no. 3, pp. 383–392, 1990.
- [34] D. A. Winter, A. E. Patla, F. Prince, M. Ishac, and K. Gielo-perczak, "Stiffness control of balance in quiet standing," *Journal of Neurophysiology*, vol. 80, no. 3, pp. 1211–1221, 1998.
- [35] A. D. Kuo and J. M. Donelan, "Dynamic Principles of Gait and Their Clinical Implications," *Physical therapy*, vol. 90, no. 2, pp. 157–174, 2010.
- [36] "The Average Walking Stride Length," <https://livehealthy.chron.com/average-walking-stride-length-7494.html>, accessed on 2020-03-03.
- [37] "Friction in contact between rotating bodies - MATLAB - MathWorks Benelux," <https://nl.mathworks.com/help/phymod/simscape/ref/rotationalfriction.html>, accessed on 2020-03-20.

VIII. ACKNOWLEDGEMENTS

I want to thank dr. Thomas Geijtenbeek and prof.dr. Frans van der Helm for guiding me and providing valuable insights during the research and dr.ir. Riender Happee for reviewing this article. Lastly, I want to thank the MARCH exoskeleton group for providing the data and SolidWorks files to do all simulations as an exemplary test case. None of the presented work is directly related with the work done by the MARCH exoskeleton group.

APPENDIX I
OPTIMIZED PARAMETERS

TABLE VI: Early stance phase parameters

	<i>init</i>	Hip AA	<i>init</i>	Hip FE	<i>init</i>	Knee	<i>init</i>	Ankle
global/local		global		global		local		local
ϕ [deg]	0	-0.00829	-5	-7.13661	0	0.68973	10	6.751537
K_p [-]	0.1	0.005204	0.5	0.900534	0.4	0.685663	0.2	0.146513
K_d [-]	0.01	0.021679	0.01	0.010034	0.01	0.041037	0.01	-0.00231

TABLE VII: Late stance phase parameters

	<i>init</i>	Hip AA	<i>init</i>	Hip FE	<i>init</i>	Knee	<i>init</i>	Ankle
global/local		global		global		local		local
ϕ [deg]	3	-0.22987	-5	-6.91773	0	-0.34453	-5	6.933055
K_p [-]	0.3	0.424511	0.1	0.131978	0.2	0.033842	0.3	0.547605
K_d [-]	0.01	0.004356	0.01	0.005729	0.01	0.013995	0.01	0.028713

TABLE VIII: Liftoff phase parameters

	<i>init</i>	Hip AA	<i>init</i>	Hip FE	<i>init</i>	Knee	<i>init</i>	Ankle
global/local		global		global		local		global
ϕ [deg]	-5	-4.24232	-15	-14.9193	70	-83.306	-5	-3.78197
K_p [-]	0.1	0.025099	0.1	0.070739	0.2	0.490179	0.8	0.612985
K_d [-]	0.01	0.001187	0.01	0.027355	0.01	0.001005	0.01	0.004394

TABLE IX: Swing phase parameters

	<i>init</i>	Hip AA	<i>init</i>	Hip FE	<i>init</i>	Knee	<i>init</i>	Ankle
global/local		global		global		local		local
ϕ [deg]	-4	-7.49878	80	95.32622	-60	-52.6539	5	0.293716
K_p [-]	0.5	0.336898	0.5	0.015871	0.3	0.486089	0.2	0.273894
K_d [-]	0.01	0.010408	0.01	0.00074	0.04	0.015301	0.06	0.038348
K_ϕ [-]	0.8	0.681578	0.1	0.006481				

TABLE X: Landing phase parameters

	<i>init</i>	Hip AA	<i>init</i>	Hip FE	<i>init</i>	Knee	<i>init</i>	Ankle
global/local		global		global		global		global
ϕ [deg]	-3	-5.58831	7	8.073045	-5	-12.4379	0	2.702169
K_p [-]	0.5	0.734842	0.2	0.13002	0.5	0.452061	0.2	0.082435
K_d [-]	0.01	0.014387	0.01	0.035432	0.01	0.033731	0.01	0.004885
K_ϕ [-]	0.2	0.154087	0.1	0.029558				

TABLE XI: Standing balance parameters

	<i>init</i>	Hip AA	<i>init</i>	Hip FE	<i>init</i>	Knee	<i>init</i>	Ankle
global/local		global		global		global		global
ϕ [deg]	0	0.0003521	-5	-9.1618314	5	0.79111089	-10	-0.97446231
K_p [-]	0.3	3.0815752	0.8	1.371926	0.6	1.1138292	0.25	0.4389075
K_d [-]	0	0.4903554	0	0.008149795	0	0.021671139	0	0.023554345

APPENDIX II
 ADDITIONAL RESULTS ON LOCOMOTION

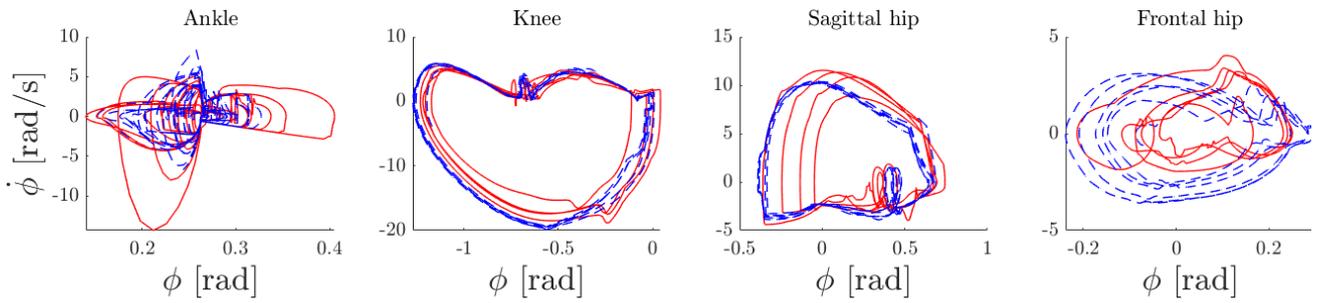


Fig. 26: After a forward push of 300 N for 0.1 s, the model takes three steps to return to the limit cycle, which can be seen in hip behaviour. The red part illustrates the corrected steps.

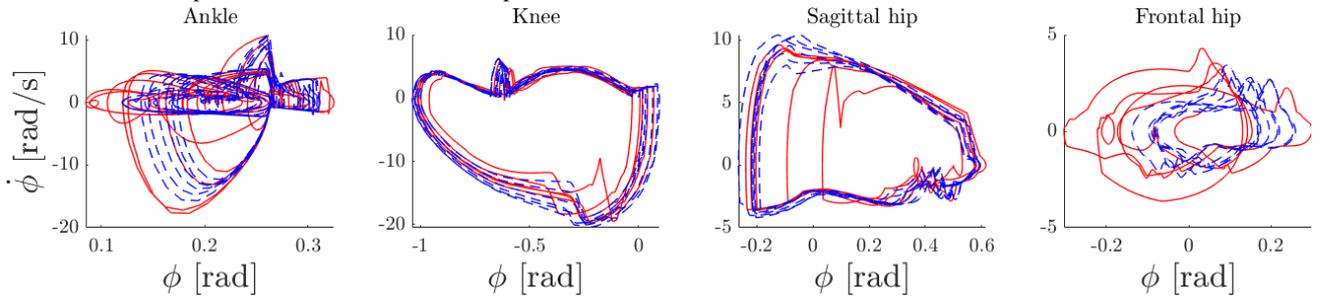


Fig. 27: After a sideways push of 300 N for 0.1 s, the model takes two steps before returning to the limit cycle, which can be seen in hip behaviour. The red part illustrates the corrected steps.

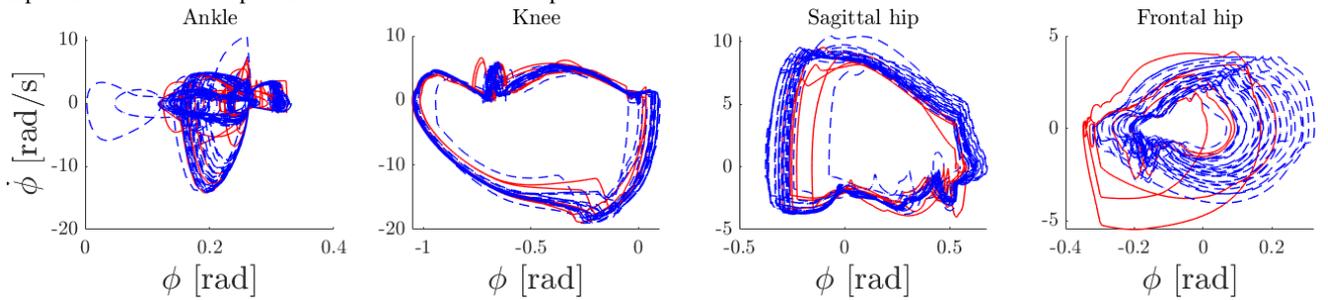


Fig. 28: After a sideways push of 400 N the model doesn't return to its limit cycle. The hip abductors keep on increasing the step-width, resulting in a sideways gait. The red part marks the push, the blue part starts 3 seconds later.

APPENDIX III
STANDING BALANCE POSITION

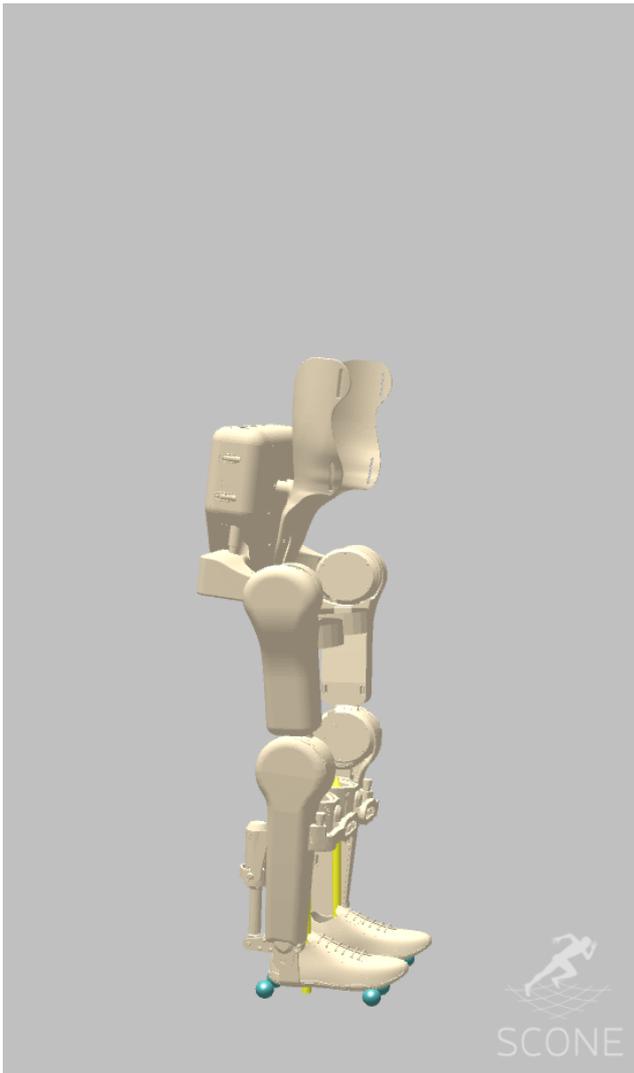


Fig. 29: The resulting pose in which the model exhibits standing balance.