2018 7th IEEE International Conference on Biomedical
Robotics and Biomechatronics (Biorob)
Enschede, The Netherlands, August 26-29, 2018

A Conceptual High Level Controller to Walk with Active Foot Prostheses/Orthoses

Mahdy Eslamy, Arndt F. Schilling

***Abstract*— In this paper, the goal is to develop a high level controller for active prosthetic feet that can continuously estimate the ankle motion based on the shank motion. The proposed controller does not require speed determination, gait percent identification, input data manipulation, look-up tables or switching rules. To do this, the Gaussian process (GP) regression is used. The performance of the controller has been tested for walking speed of 0.6, 0.9, 1.2, 1.4 and 1.6 m/s. The results showed that the controller had lower estimation quality when input was only shank angular velocity or shank angle. However, the aggregated angular velocity and angle input resulted in high output estimation quality. Furthermore, for each speed, the estimation quality was more acceptable when the controller was trained for it. Accordingly, when the high level controller was tested without previous training, the estimation quality was less acceptable.**

I. Introduction

Currently, the transtibial amputees mostly use passive or quasi-passive foot prostheses. These devices are not able to emulate completely the required ankle kinematics and kinetics during different speeds and gaits. In contrast, an intact human ankle is able to produce net positive work during stance [1] to provide the required push-off power for a wide range of speeds and gaits.

To improve amputee ambulation, active foot prostheses are under development [2–4]. Such active devices are expected to produce net positive work and emulate the required human foot motion more accurately. Nevertheless, the broad commercialization of active prosthetics is hindered due to different challenges.

One of the main issues is the design of the high level controller for such devices. The high level controller is in charge to estimate the desired operation of the actuator in line with the locomotion of the amputee user. The output of the high level controller would then be the input to the low level controller to finally create the command signal for the motor.

To date, different high level controllers have been suggested for active prosthetic feet. In [3] to control an active ankle prosthesis, the gait cycle was divided to different sections based on the angle-torque data. In this method, the actuator operation changes based on if-then decision making stations within a single stride. A similar state machine approach was used in [5], in which the gait cycle was divided to different sections based on the kinematics and kinetics data from ankle, knee and shank. For each section, the ankle impedance characteristics were previously determined through experiments. These stiffness and damping gains were used together with the online kinematic data from the sensors to form the command signal for the motor.

In contrast to those discrete approaches, a semi-continuous procedure was suggested in [6] to estimate the desired motor positions. In this phase plane method, shank angle and angular velocity were used to determine the speed and gait percent. Then, based on a look-up table, the corresponding desired motor positions was found out in accordance with that speed and gait percent. In this method, it was not required to divide the gait to different sections, however, to determine the speed it was required to use if-then within the controller program.

In addition to mechanical sensors, the myoelectric EMG signals [7–9] were also used in active foot prostheses. In [10] it was shown that sensor fusion approach (EMG + mechanical) can result in better prosthesis control compared to the methods that use only EMG signals or only the mechanical sensors. In addition, Support Vector Machines (SVM) [10], Gaussian Mixture Models (GMM) [11, 12], Linear Discriminant Analysis (LDA) [13, 14], Neural Networks [13], Dynamic Bayesian Networks [15] and k-nearest neighbor (k-NN) [16] are also other methods that have been used to process the sensory inputs and estimate the desired actuator (usually a motor) status as output.

In this paper, the goal is to develop a high level controller for active prosthetic feet that can continuously estimate the ankle motion based on the shank motion. The proposed controller does not require speed determination, gait percent identification, input data manipulation or switching rules as opposed to the methods suggested by [2–4]. To do this, the Gaussian process (GP) regression is used.

The paper structure is organized as follows. First, we explain the definitions, required information and the fundamentals and describe the function-based viewpoint used in this study. Next, a string of different scenarios are planned and the performance of the proposed high level controller is evaluated for each scenario. At the end, final conclusions are made together with the suggestions for future works and possible improvements.

II. Method

A. Fundamentals and Basic Definitions

In Fig. 1, the general concept of the control structure for active prosthetic feet is presented. As observed, it is comprised of a high level controller and a low level controller. The high level controller estimates desired motor positions y

Applied Rehabilitation Technology ART Lab, Department for Trauma Surgery, Orthopaedics and Plastic Surgery, Universitätsmedizin Göttingen (UMG), 37075, Göttingen, Germany, {mahdy.eslamy, arndt.schilling}@med.uni-goettingen.de

978-1-5386-8183-1/18/$31.00 ©2018 IEEE

in line with the user's ambulation. The low level (the motor) controller (e.g. a PD controller or an impedance controller) then creates an error signal $e = y - y_a$ based on the estimated value y and the actual value y_a from motor sensors. The error is further processed using gains to finally create a command signal to actuate the motor.

In human biomechanics, the gait cycle starts with the heel contact and ends with the next contact of the same foot. A gait cycle is divided to one hundred sections called gait percent [1]. Each speed and each gait percent has its corresponding desired motor position that should be mimicked by an active prosthesis. Basically, the task of the high level controller is to estimate a desired motor position corresponding to each speed and gait percent (see Fig. 1). For each speed, the expected desired motor positions could be obtained offline based on the ankle angles, torques, the geometry of the prosthetic device and the stiffness of the series spring. See [2, 17, 18] for fully detailed discussions on how to obtain expected desired values offline. These offline values are required in order to compare them later with the estimated ones.

As seen in Fig. 1, in order to develop the high level controller, some input is required. For this study, we've collected data for shank angular velocity $\dot{\theta}_{sh}$ for walking 0.6, 0.9, 1.2, 1.4 and 1.6 $\frac{m}{s}$. The data is related to a healthy male subject (1.86 m, 76 kg, measurement through: LPY550AL, STMicroelectronics, 1 kHz).

One possible approach to have another input without adding more sensors, is to integrate from shank angular velocity. Hence, two inputs would be available i.e. $\dot{\theta}_{sh}$ and θ_{sh}. In this way it is possible to investigate the performance of the proposed high level controller with only one input $x = \dot{\theta}_{sh}$ or $x = \theta_{sh}$ or two combined inputs as $x = [\dot{\theta}_{sh}, \theta_{sh}]$.

For the integration of shank angular velocity $\dot{\theta}_{sh}$, using a standard integration approach such as $\frac{1}{s}$ resulted in drift. To overcome this, a pseudo-integrator as $\frac{1}{s+0.1}$ was used to have stable θ_{sh}. Although this would not produce the true shank angles, it does not pose problem for the high level controller. The proposed high level controller only requires a set of inputs that can be related to a specific speed. In other words, the controller does not depend on the true physical meaning of the input parameters. This approach is different from the methods that work based on the state-machines, where a true real value for angle or torque is used to switch between the states [3, 5].

B. GP-based High Level Controller

In this paper, the problem of developing the high level controller is converted into the problem of finding an appropriate function f that can continuously map the inputs x ($\dot{\theta}_{sh}$ or θ_{sh} or both) to the outputs y (motor positions). Based on a functional point of view, this can be stated as $y = f(x)$.

To define function f, the Gaussian process (GP) regression [19] has been used. Regression in general deals with the prediction of continuous quantities (e.g. for our work motor positions y). In fact, the goal is to make inferences about the relationship between the inputs x and outputs y. Furthermore,

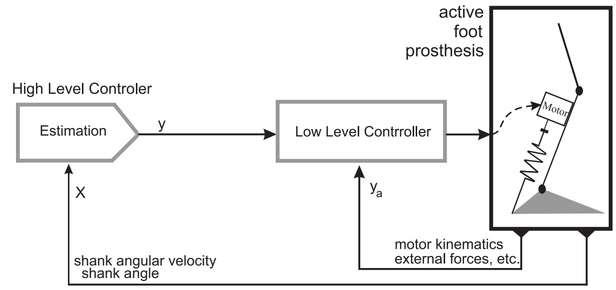

Fig. 1. An overall control structure for an active foot prosthesis. A part of the acquired sensory information (X) is used by the high level controller to estimate and predict the desired actuator status (e.g. desired motor positions y). Furthermore, some part of the sensory data (y_a) is used to create the required command signal through the low level (motor) controller to actuate it. Note that different actuation mechanisms could be used for the active foot prostheses [17], however in this study the results are related to the series elastic actuator (SEA). Read section II for detailed information.

the GP Regression aims at learning a function f that can be used to estimate outputs for the new (unseen test) inputs [19, 20].

A GP is defined by its mean $m(x)$ and covariance function $k(x, x')$ (for the input pair x and x') as [19]

$$f(x) \sim \mathcal{GP}(m(x), k(x, x')) \tag{1}$$

where $m(x) = \mathrm{E}[f(x)]$ (E denotes expectation of $f(x)$) and $k(x, x') = \mathrm{E}[(f(x) - m(x))(f(x') - m(x'))]$. To use GP for prediction, some priori training is required. The training input could be $[\dot{\theta}_{sh}]$ or $[\theta_{sh}]$ or $[\dot{\theta}_{sh}, \theta_{sh}]$ and could be formed like $D = \{(x_i, f_i) | i = 1, \ldots, n\}$. Therefore, it will consist of n observations, where x_i is a d-dimensional input vector and f_i (or in other words y_i) is the corresponding function (output) value (therefore $y_i = f(x_i)$). In addition, concatenating the training observations would form the aggregated input matrix X as

$$X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix}, \quad y = f = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix} \tag{2}$$

To evaluate the performance of GP in terms of the estimation (prediction), a test set is also required. The test inputs are denoted with x^*. Accordingly, the outputs related to x^* are marked with f^* (or in fact $y^* = f^*(x^*)$).

It is logical to think that inputs close to each other would have quite similar output values and therefore the training inputs that are close to the test inputs should be logically informative for prediction. In GP, the covariance function k defines such a similarity. The covariance function is also a designer-defined parameter. For GP, it is possible to select different functions [19, 20]. In this study we evaluated different functions, one of them was Matérn covariance function. The Matérn class of covariance function is in the form of

$$k_{Matérn}(r) = \frac{2^{1-\nu}}{\Gamma(\nu)} (\frac{\sqrt{2\nu}r}{l})^{\nu} K_{\nu}(\frac{\sqrt{2\nu}r}{l}) \tag{3}$$

r is the the absolute difference of input pairs x and x' ($r = |x - x'|$), ν and l are positive parameters called hyperparameters Θ, where K_ν is a modified Bessel function [19]. One common approach to determine the hyperparameters is to maximize the log marginal likelihood [19, 20]

$$logp(y|X,\Theta) = -\frac{1}{2}(y^T K(X,X)y + log|K(X,X)| + nlog2\pi) \tag{4}$$

in which $p(A|B)$ means conditional probability of A given that event B is true. The first term of Eq. 4 addresses data fit, the second term introduces a complexity penalty and the last term is a normalization constant. The log marginal likelihood automatically performs a trade-off between model fit and complexity. Several optimization algorithms could be used. In this study hyperparameters were optimized by different approaches one of them was exact inference method together with Gaussian Likelihood.

The joint distribution of the training outputs f and the test outputs f^* would be [19]

$$\begin{bmatrix} f \\ f^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X,X) & K(X,X^*) \\ K(X^*,X) & K(X^*,X^*) \end{bmatrix}\right) \tag{5}$$

where $K(X, X^*)$ is the $n \times n^*$ matrix of the covariances evaluated at all pairs of training and test points. A $K(X, X)$ is defined as

$$K(X,X) = \begin{pmatrix} k(x_1,x_1) & k(x_1,x_2) & \cdots & k(x_1,x_n) \\ k(x_2,x_1) & k(x_2,x_2) & \cdots & k(x_2,x_n) \\ \vdots & \vdots & \vdots & \vdots \\ k(x_n,x_1) & k(x_n,x_2) & \cdots & k(x_n,x_n) \end{pmatrix} \tag{6}$$

A similar statement could be expected for other $K(.,.)$. $\mathcal{N}$ denotes normal distribution.

In light of Eq. 5, it is possible to obtain information for f^* on the condition of having X, f and X^* as

$$f^*|X, f, X^* \sim \mathcal{N}(\bar{f^*}, cov(f^*)), \text{ where} \tag{7a}$$

$$\bar{f^*} = K(X^*,X)K(X,X)^{-1}f \tag{7b}$$

$$\text{cov}(f^*) = K(X^*,X^*) - K(X^*,X)K(X,X)^{-1}K(X,X^*) \tag{7c}$$

where $\bar{f^*}$ and $\text{cov}(f^*)$ are GP posterior mean and GP posterior covariance respectively. Eqs. 7 provide the solution for f^* to estimate (predict) the desired outputs related to the unseen new test inputs (i.e. the test inputs X^*). Consequently, it has the potentials to be used for developing the high level controller for active foot prosthesis.

For each of those five speeds, 70% of the collected data set was used for the training and the rest was used for testing the performance of the proposed controller.

Note that there is no gold standard to define which estimation quality should be deemed as acceptable and which one as not acceptable. However, in order to have a tool to draw conclusions about the impact of this study, we use the criterion that was used in [21] for the acceptable estimation quality. In that study, R^2 values higher than 0.8 were a sign of acceptable estimations (the maximum is 1). Therefore, we will also use that definition to define whether estimation qualities are acceptable or not. Therefore, in Tab. I, R^2 values higher than 0.8 are highlighted by green color and those values less than 0.8 are highlighted by orange color. This facilitates a better visualization of the results.

III. Results

The GP provided the required tool to develop a continuous mapping from the inputs to the outputs without the need for speed determination, gait percent identification, look-up tables and switching rules. Now, it is time to determine which input type results in the best estimation quality for this tool. The following scenarios clarify the answer to this issue. At first, we evaluate the estimation quality with minimum number of inputs i.e. input is only the shank angular velocities or the shank angles for each speed.

A. single input: shank angular velocity

In this scenario, the estimation quality is investigated when the input is the shank angular velocity [$\dot{\theta}_{sh}$]. The corresponding result is shown in Fig. 2 (A) and Tab. I.

As it is seen in Tab. I, for all speeds the results are not acceptable.

B. single input: shank angle

For this scenario, the input is the shank angle [θ_{sh}]. The corresponding results are shown in Fig. 2 (B) and Tab. I.

As it is seen in Tab. I, in general, the performance is better than the previous scenario. Nonetheless, the results are still not acceptable.

C. double-input: shank angular velocity and angle [$\dot{\theta}_{sh}$,θ_{sh}]

In this scenario, the inputs are the shank angular velocity and angle [$\dot{\theta}_{sh}$,θ_{sh}]. The corresponding results are shown in Fig. 2 (C) and Tab. I.

As it is seen, for this scenario, the estimation quality is acceptable. For 0.6, 1.2 and 1.4 m/s the estimations are close to perfect.

As it is seen the double-input approach outperformed the single-input scenarios. Having this result in mind, it is of interest to see if it is possible to still have acceptable estimations while the high level controller is trained with fewer number of speeds. In the following scenarios this point will be under investigation.

1) input: [$\dot{\theta}_{sh}$,θ_{sh}], one speed for training:

In this scenario, the inputs are the shank angular velocity and angle [$\dot{\theta}_{sh}$,θ_{sh}]. However, the high level controller is trained with the data set related to 0.6 m/s. The performance of the controller is then tested for all five speeds. The corresponding result is shown in Fig. 2 (D) and Tab. I.

As it is seen in Tab. I, the performance of the proposed controller was acceptable only for the speed that it was trained (0.6 m/s). For other speeds that had no training procedure, the estimation quality declined (in comparison to subsection III-C).

2) input: [$\dot{\theta}_{sh}$,θ_{sh}], two speeds for training:

In this scenario, the inputs are the shank angular velocity and angle [$\dot{\theta}_{sh}$,θ_{sh}]. However, the high level controller is trained with the data set related to 0.6 and 0.9 m/s. The performance of the controller is then tested for all speeds. The corresponding result is shown in Fig. 2 (E) and Tab. I.

As it is seen, similar to previous subsection, the performance of the controller was acceptable only for the speeds it was trained. For other speeds the results were not as acceptable as before according to Tab. I (in comparison to subsection III-C).

3) input: [$\dot{\theta}_{sh}$,θ_{sh}], three speeds for training:

In this scenario, the high level controller is trained with the data set related to 0.6, 0.9 and 1.2 m/s. The performance of the controller is then tested for all speeds. The corresponding result is shown in Fig. 2 (F) and Tab. I. Similar to the previous results, the performance of the controller was near to perfect only for the speeds it was trained. For other speeds the results were not as acceptable as before according to Tab. I.

4) input: [$\dot{\theta}_{sh}$,θ_{sh}], four speeds for training:

For this scenario, the high level controller is trained

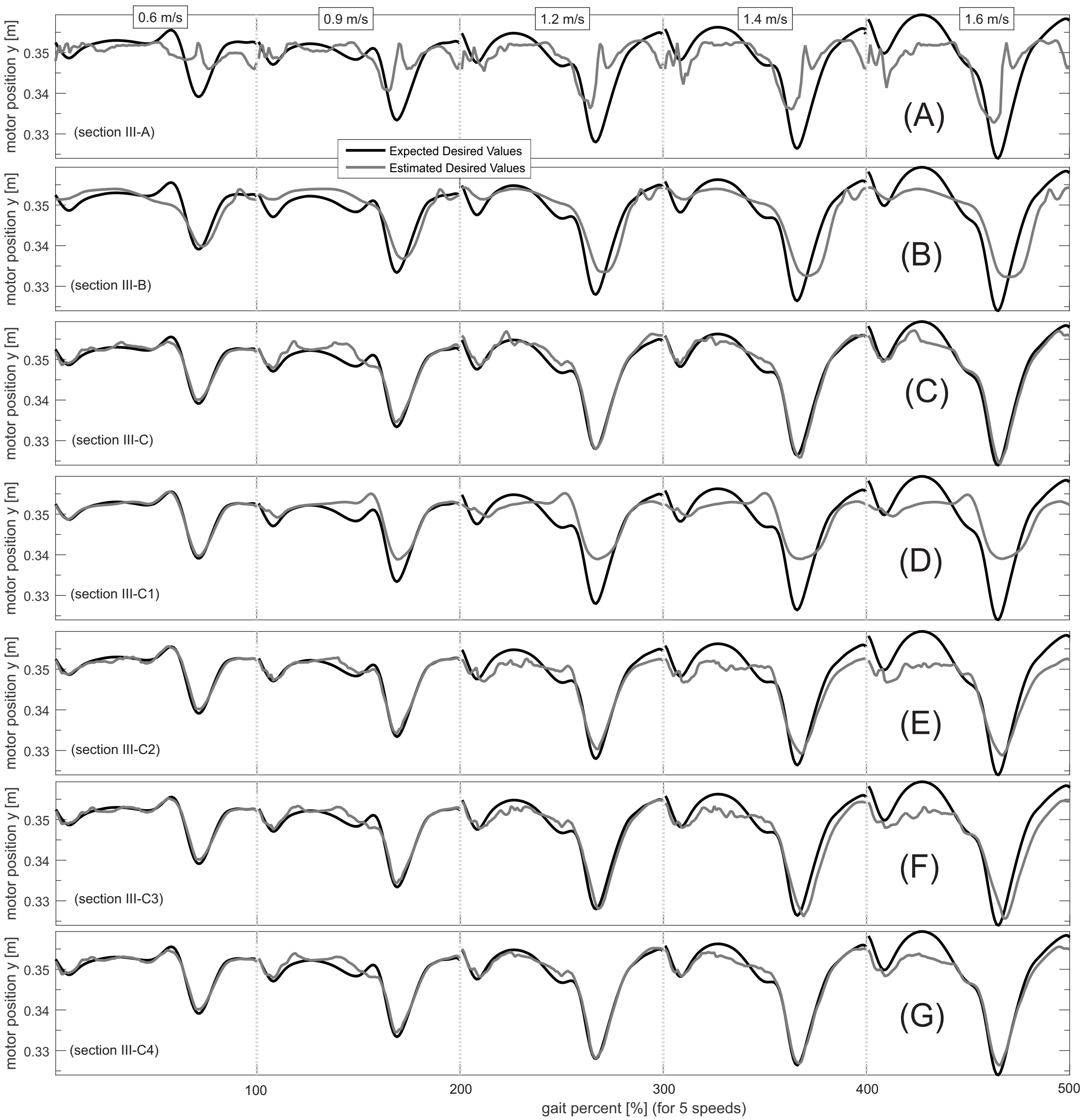

Fig. 2. (related to section III) The estimated and expected desired motor positions for five speeds (see also Tab. I).

TABLE I

THE ROOT MEAN SQUARE (RMS) ERROR, THE MAXIMUM (ABSOLUTE) ERROR AND R^2 VALUES FOR DIFFERENT SPEEDS AS DISCUSSED IN SECTION III

input scenario ↓	speed [m/s] →	0.6	0.9	1.2	1.4	1.6
$[\dot{\theta}_{sh}]$ (section: III-A)	RMS error $[m]$	3.8e-3	4.6e-3	5.7e-3	5.8e-3	7.1e-3
	maximum (absolute) error $[m]$	13.3e-3	18.6e-3	22.6e-3	20.6e-3	22.5e-3
	R^2	0.07	0.19	0.38	0.49	0.46
$[\theta_{sh}]$ (section: III-B)	RMS error $[m]$	2.2e-3	2.7e-3	3.1e-3	3.9e-3	5.4e-3
	maximum (absolute) error $[m]$	5.6e-3	5.7e-3	8.2e-3	9.5e-3	11.4e-3
	R^2	0.68	0.72	0.82	0.77	0.70
$[\dot{\theta}_{sh}, \theta_{sh}]$ (section: III-C)	RMS error $[m]$	0.7e-3	1.8e-3	1.3e-3	1.3e-3	2.1e-3
	maximum (absolute) error $[m]$	1.5e-3	4.9e-3	2.6e-3	2.9e-3	5e-3
	R^2	0.97	0.87	0.97	0.98	0.95
$[\dot{\theta}_{sh}, \theta_{sh}]$ (section: III-C.1)	RMS error $[m]$	0.4e-3	2.6e-3	4.2e-3	4.6e-3	6.1e-3
	maximum (absolute) error $[m]$	0.7e-3	5.7e-3 11.0	e-3 12.7	e-3	15.3e-3
	R^2	0.99	0.75	0.67	0.67	0.61
$[\dot{\theta}_{sh}, \theta_{sh}]$ (section: III-C.2)	RMS error $[m]$	0.6e-3	0.9e-3	2.2e-3	3.2e-3	5e-3
	maximum (absolute) error $[m]$	1.8e-3	2.7e-3	4.1e-3	6.3e-3	8.8e-3
	R^2	0.97	0.97	0.91	0.85	0.74
$[\dot{\theta}_{sh}, \theta_{sh}]$ (section: III-C.3)	RMS error $[m]$	0.6e-3	1.3e-3	1.7e-3	3e-3	4.7e-3
	maximum (absolute) error $[m]$	1.5e-3	3.1e-3	3.6e-3	5.9e-3	8.5e-3
	R^2	0.98	0.93	0.95	0.86	0.76
$[\dot{\theta}_{sh}, \theta_{sh}]$ (section: III-C.4)	RMS error $[m]$	0.6e-3	1.4e-3	1e-3	1.5e-3	3e-3
	maximum (absolute) error $[m]$	1.4e-3	3.2e-3	2.6e-3	3.2e-3	6.2e-3
	R^2	0.98	0.92	0.98	0.97	0.90

with the data set related to 0.6, 0.9, 1.2 and 1.4 m/s. The performance of the controller is then tested for all speeds. The corresponding result is shown in Fig. 2 (G) and Tab. I. As it seen again, the performance of the controller was acceptable for those speeds it was trained. For the untrained speed of 1.6 m/s, the results were not as acceptable as before according to Tab. I.

IV. DISCUSSIONS, CONCLUSION & SUGGESTIONS

As seen, the two-input scenario outperformed the single-input scenario. One explanation for this observation could be the relationship between expected desired motor positions and the shank angular velocity or shank angle. This has been shown in Fig. 3. As it is seen, for each shank angular velocity or angle, (at least) two values for desired motor position can be obtained. This has the potential to put the high level controller under difficulty to estimate the acceptable output for a specific input.

In addition, as it was seen from Fig. 2 and Tab. I, for the untrained speeds, the high level controller was not able to estimate as strongly as the case it received training (e.g. compare the results in section III-C with subsection III-C.1).

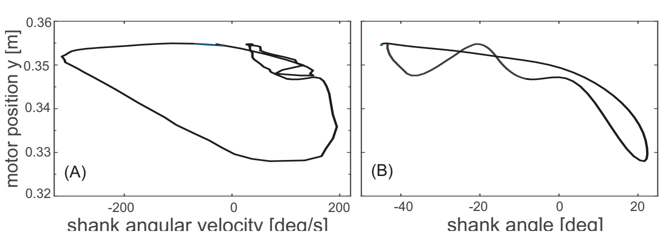

Fig. 3. The diagrams of the expected desired motor positions with respect to (A) shank angular velocity and (B) shank angle (diagrams for speed of 1.2 m/s).

This was observed both in extrapolation and interpolation investigations.

This can have different reasons. One might be that the number of the inputs were not enough (i.e. more than two input types might be required). Another reason might be the type of the employed inputs, e.g. a fusion of those kinematic inputs and EMG signals might have better results in this regard. In addition, the method that was used for designing the high level controller might be another reason, since GP is a supervised machine learning approach.

Regarding the training with only some speeds, e.g. one speed (subsection III-C.1), a question that might arise is that why that specific speed was used for training and not another speed e.g. 1.2 m/s was used for training. As a matter of fact, different combinations of speeds were investigated for this purpose (for III-C.1 to III-C.4). At the end, the result was that the estimation was acceptable for the trained speeds and for untrained speeds the estimation quality declined. Although the results might be still acceptable according to the criterion we used in this study.

In this study, we showed the feasibility of developing a high level controller for active foot prosthesis using shank angular velocity and angle. This was achieved through GP regression. We are currently conducting parallel investigations in order to find out the pros and cons of the GP-based high level controllers. In addition, we are investigating if this machine learning approach could be enhanced through combining with other techniques, so that the high level controller could be extended for a variety of locomotion types.

V. ACKNOWLEDGEMENT

The required data for this work was obtained from Lauflabor Laboratory, Technical University of Darmstadt, Germany.

This work was partially supported by the Grant from the Bundesministerium für Bildung und Forschung (BMBF) to A. F. Schilling (INOPRO-16SV7656).

References

[1] M. W. Whittle, *Gait analysis: an introduction*, 2003.

[2] K. W. Hollander, R. Ilg, T. Sugar, and D. Herring, "An efficient robotic tendon for gait assistance," *ASME Jour. of. Biomech. Eng.*, vol. 128, pp. 788–791, 2006.

[3] S. K. Au, J. Weber, and H. Herr, "Powered ankle-foot prosthesis improves walking metabolic economy," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 51–66, 2009.

[4] F. Sup, H. Varol, J. Mitchell, T. Withrow, and M. Goldfarb, "Self-contained powered knee and ankle prosthesis: Initial evaluation on a transfemoral amputee," in *IEEE Int'l Conf. on Rehab. Robo.*, 2009, pp. 638–644.

[5] F. Sup, H. A. Varol, J. Mitchell, T. Withrow, and M. Goldfarb, "Design and control of an active electrical knee and ankle prosthesis," in *IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, 2008, pp. 523–528.

[6] M. Holgate, T. Sugar, and A. Bohler, "A novel control algorithm for wearable robotics using phase plane invariants," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 3845–3850.

[7] S. Au, P. Bonato, and H. Herr, "An emg-position controlled system for an active ankle-foot prosthesis: an initial experimental study," in *IEEE International Conference on Rehabilitation Robotics ICORR*, 2005, pp. 375–379.

[8] D. P. Ferris, K. E. Gordon, G. S. Sawicki, and A. Peethambaran, "An improved powered ankle–foot orthosis using proportional myoelectric control," *Gait & posture*, vol. 23, no. 4, pp. 425–428, 2006.

[9] J. R. Koller, C. D. Remy, and D. P. Ferris, "Comparing neural control and mechanically intrinsic control of powered ankle exoskeletons," in *International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 2017, pp. 294–299.

[10] H. Huang, F. Zhang, L. J. Hargrove, Z. Dou, D. R. Rogers, and K. B. Englehart, "Continuous locomotion-mode identification for prosthetic legs based on neuromuscular–mechanical fusion," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 10, pp. 286–2875, 2011.

[11] H. Varol, F. Sup, and M. Goldfarb, "Multiclass real-time intent recognition of a powered lower limb prosthesis," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 3, pp. 542–551, 2010.

[12] A. Kilicarslan, S. Prasad, R. G. Grossman, and J. L. Contreras-Vidal, "High accuracy decoding of user intentions using eeg to control a lower-body exoskeleton," in *Engineering in medicine and biology society (EMBC), 2013 35th annual international conference of the IEEE*. IEEE, 2013, pp. 5606–5609.

[13] H. Huang, T. Kuiken, and R. Lipschutz, "A strategy for identifying locomotion modes using surface electromyography," *IEEE Transactions on Biomedical Engineering*, vol. 56, pp. 65–73, 2009.

[14] A. J. Young, A. M. Simon, N. P. Fey, and L. J. Hargrove, "Classifying the intent of novel users during human locomotion using powered lower limb prostheses," in *International IEEE/EMBS conference on Neural engineering (NER)*. IEEE, 2013, pp. 311–314.

[15] A. J. Young, A. M. Simon, N. P. Fey, and L. J. Hargrove, "Intent recognition in a powered lower limb prosthesis using time history information," *Annals of biomedical engineering*, vol. 42, no. 3, pp. 631–641, 2014.

[16] H. Varol and M. Goldfarb, "Real-time intent recognition for a powered knee and ankle transfemoral prosthesis," in *International Conference on Rehabilitation Robotics ICORR*, 2007, pp. 16–23.

[17] M. Eslamy, M. Grimmer, and A. Seyfarth, "Effects of unidirectional parallel springs on required peak power and energy in powered prosthetic ankles: Comparison between different active actuation concepts," *IEEE Int'l Conf. on Robo. Biomim.*, pp. 2406–2412, 2012.

[18] M. Eslamy, M. Grimmer, S. Rinderknecht, and A. Seyfarth, "Does it pay to have a damper in a powered ankle prosthesis? a power-energy perspective," *IEEE Int'l Conf. on Rehab. Robo.*, pp. 1–8, 2013.

[19] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[20] [Online]. Available: http://www.gaussianprocess.org/gpml

[21] R. A. Bogey and L. A. Barnes, "An emg-to-force processing approach for estimating in vivo hip muscle forces in normal human walking," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 8, pp. 1172–1179, 2017.