Gait Trajectory and Event Prediction from State Estimation for Exoskeletons during Gait

Kevin Tanghe[1], Friedl De Groote[2], Dirk Lefeber[3], Joris De Schutter[1], Erwin Aertbeliën[1]

***Abstract*—A real-time method is proposed to obtain a single, consistent probabilistic model to predict future joint angles, velocities, accelerations and jerks, together with the timing for the initial contact, foot flat, heel off and toe off events. In a training phase, a probabilistic principal component model is learned from normal walking, which is used in the online phase for state estimation and prediction. This is validated for normal walking and walking with an exoskeleton. Without exoskeleton, both joint trajectories and gait events are predicted without bias. With exoskeleton, the trajectory prediction is unbiased, but event prediction is slightly biased with a maximum of 33 ms for the toe off event. Performance is compared with predictions based on only the population mean. Without exoskeleton, estimation errors are 5 to 30% lower with our method. With exoskeleton, trajectory prediction errors are up to 20% lower, but gait event prediction errors only improve for foot flat (30%) and are worse for other events (30%–50%). The ability to predict future joint trajectories and gait events offers opportunities to design exoskeleton controllers which anticipate these trajectories and events, allowing better tracking control and smoother, accurately timed transitions between different control modes.**

***Index Terms*—State estimation, kinematics prediction, event prediction, real time, exoskeleton**

I. Introduction

LOWER-limb exoskeletons have the potential to restore or increase the mobility of impaired subjects. Some exoskeletons are fixed to the environment and are used for rehabilitation purposes, e.g., Lopes [1] and Lokomat [2]. Other exoskeletons allow the subject to move freely, and can be used for rehabilitation or permanent assistance [3]–[5].

The control of an exoskeleton can be enhanced if information on the future is available. For instance, if the future joint trajectory is known, adding a feedforward term will result in better tracking behavior than relying purely on feedback. The feedforward term can also be used to compensate time delays in the control. Another example is the control mode switching during gait. Gait has different phases. Each phase is linked with a different exoskeleton control mode. For instance, changing the ankle angle requires higher torques in the stance phase than in the swing phase since the ankle has to push up the whole exoskeleton and human body during the stance phase. The control parameters of the ankle joint will therefore change between discrete control modes depending on the current phase. It is beneficial to have a smooth transition between control modes instead of a discrete switch. This requires information on the future such that the transition to another control mode can be initiated prior to the actual occurrence of the gait event that marks the change of gait phase.

To determine the amount of assistance, most of the current exoskeletons rely on a reference trajectory as an input [6], [7]. The controller will assist the subject to follow this trajectory. These reference trajectories can be calculated in a static way, but for the user it would feel more natural if this reference trajectory is constantly adapting to the subject's current execution style. Hence, estimated future healthy joint trajectories can be used as a reference input.

The paragraphs above motivate the two goals of this paper: 1) the prediction of future joint trajectories during gait and 2) the prediction of future gait events. Section II gives an overview of the state of the art and states the contributions of this paper. Section III reports the collected data. Section IV explains our method. Section V presents validation experiments and section VI discusses the results of our method.

II. Contributions to the State of the Art

A. Trajectory Prediction

The prediction of joint trajectories has already been tackled by several researchers, mainly in the rehabilitation research field. In this field, patients are guided towards an estimated reference trajectory. These trajectories used to be recordings of healthy gait or were set by a clinical expert. Nowadays methods are available to make these trajectories more patient specific. One possibility is to estimate spatio-temporal values of key points in the gait cycle, such as extrema of the joint angle trajectories, based on a regression of different parameters (e.g. walking speed, body height, step length, ...) [8], [9]. Complete trajectories are reconstructed by spline interpolation between these key points. It is also possible to estimate the total trajectory (instead of some key points) using Gaussian process regression on a large set of body parameters [10]. Yet another approach is to collect a dataset with reference joint trajectories and body parameters (age, height, mass, ...) from different subjects. A reference trajectory for a new subject can be estimated by selecting trajectories from subjects with similar body parameters and apply principal component analysis on the selected trajectories [11]. In this way, common features are extracted and used to create a reference trajectory for the new subject.

[1] Robotics Research Group, Department of Mechanical Engineering, KU Leuven, 3001 Heverlee, Belgium. The Robotics Research Group is a university core lab of Flanders Make. Contact: firstname.lastname@kuleuven.be

[2] Human Movement Biomechanics Research Group, Department of Movement Sciences, KU Leuven, 3001 Heverlee, Belgium.

[3] Robotics and MultiBody Mechanics Research Group (R&MM), Department of Mechanical Engineering, Vrije Universiteit Brussel, 1050 Brussels, Belgium. R&MM is a university core lab of Flanders Make.

The above methods estimate the trajectory in a static way, there is no online adaption to the execution by the subject. For better rehabilitation and comfort, joint trajectories should not be purely based on some parameters, but should adapt to the current motion of the subject. Several methods are available to adapt the offline estimated trajectories to current observations. For instance, the offline trajectory can be scaled in time and amplitude, and an offset parameter can be added [12]. The offline trajectory can also be modified online by using an admittance control law, where a feedback error term and an online learned offset are added to the reference trajectory [13]. Another approach is to set reference values for some key events, such as the step length and cycle time at heel strike and use model predictive control to generate trajectories that guide the subject to these points, given current observations [14]. An alternative for model predictive control are dynamic movement primitives [15], where an attractor guides the user towards an end goal while performing a motion which is close to some learned trajectory.

Other online adapting methods do not start from a predefined trajectory or predefined key points. For hemiparetic subjects, a trajectory for the impaired leg is generated by a low-dimensional space mapping from the observed trajectory of the healthy leg [16]. Since gait is quasi-periodic, it is also possible to get a trajectory from a weighted sum of already observed trajectories that are close to the current observations [17]. The past observations can also be used to reconstruct the quasi-periodic movement with adaptive oscillators [18]. This method is also able to estimate the current phase of the gait cycle. Furthermore, machine learning methods can predict joint angles at the next time step from a series of observed joint angles, making use of a large dataset of comparable motions [19], [20].

Although walking movements have been predicted based on a musculoskeletal model by minimizing an object function [21]–[23], this requires high computational times and is therefore not applicable in the context of real-time exoskeleton control.

B. Event prediction

Fig. 1 shows the different gait events that initiate the different gait phases. For an exoskeleton, the initial contact (IC) and toe off (TO) events are the most important, since these events determine the ground contacts. The foot flat (FF) and heel off (HO) event can also be useful, since these events mark the start and end of full ground contacts, where interaction moments are possible.

A lot of publications present algorithms for the real-time detection of the current gait phase. A good overview is presented in [24] and [25]. The algorithms use a wide variety of wearable sensor signals: foot switches, pressure insoles, accelerometers, gyroscopes, IMUs, electromyography or combinations of the above. These sensor signals are often processed with a threshold-based algorithm or a machine-learning algorithm.

In this paper, the goal is not only to detect the transition to the next gait phase as it happens, but also to predict the timing of future events. The authors are not aware of any reported methods where the timing of future events is predicted.

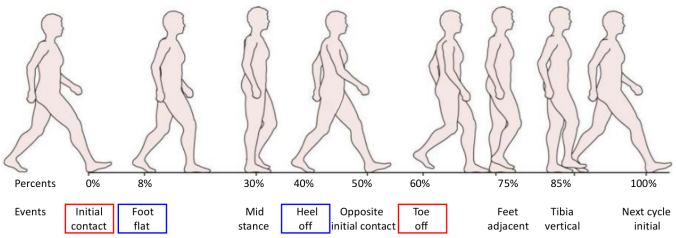

Fig. 1. One gait cycle with indicated gait events [26]. For exoskeleton purposes, the initial contact and toe off event are the most important ones. Other relevant events are foot flat and toe off.

C. Contributions

A first contribution of this work is to propose a methodology to obtain a model to predict A) smooth joint trajectories: angles, velocities, accelerations and higher order derivatives, and B) the timing of future events. This stands in contrast to the state of the art where these issues are handled separately. Prediction of future event prediction is not discussed in the state of the art. Because a single, consistent model is obtained, measurements of the joint angles also give information on the event prediction. Additionally, derivatives of the joint trajectories are calculated analytically, avoiding numerical techniques which do not produce useful results in the presence of noise. These higher order derivatives can be useful to enhance the control of assistive devices such as exoskeletons, e.g. information on the derivative of joint acceleration can improve control of series-elastic actuators [27].

Secondly, the solution presented here is probabilistic, i.e., the used model takes into account uncertainty and also generates information on the uncertainty of its predictions. We can exploit such a probabilistic model in the control of an exoskeleton: measured human motion can be checked to see whether it is probabilistically conforming to our gait model; additional model-based information can be used to improve the estimates; or control strategies can be designed to provide assistance taking into account the predicted uncertainty (e.g. see [28]). To achieve this goal we started from the probabilistic modeling approach of [29], and we adapted and extended it for the prediction of events and higher order derivatives. A new online estimation approach is developed to robustly estimate the variables in this model at run-time. Due to this added robustness, it is possible to do predictions for gait trajectories for walking with exoskeleton, whereas the model is based on demonstrations of gait without exoskeleton.

Thirdly, the method is sufficiently efficient such that it can be used in real time in a control loop.

Lastly, the developed methodology is validated on walking with and without exoskeleton and compared to predictions that only use mean trajectory data and mean event timings.

III. Data collection

Two datasets were collected. The first dataset consists of 28 healthy young subjects walking on a treadmill without

exoskeleton. The Medical Ethics Committee UZ KU Leuven and School ethics committee of the School of Healthcare Sciences of Cardiff University approved the experimental protocol. Informed consent was obtained. The walking speed varied between 2 and 5 km/h. In total, 1098 steps were observed. The second dataset consists of 5 healthy young subjects walking on a treadmill with an exoskeleton with a speed of 2 km/h. They walked in a passive condition, where the exoskeleton is unactuated, and in a transparent condition, where the exoskeleton is compensating its own dynamics [27]. The exoskeleton has six rotational joints: hip, knee and ankle joint for each leg [30] [31]. Each joint has a rotary shaft encoder to determine the position of the joint at a frequency of 500 Hz. The exoskeleton itself has no sensors to detect ground contact.

For both datasets, Vicon cameras (Vicon, Oxford) measured the trajectories of reflective markers attached to anatomical locations of the subject at a frequency of 100 Hz. A generic musculoskeletal model with 37 degrees of freedom [32], [33] was scaled to the subject's anthropometry to calculate joint kinematics from the recorded marker trajectories using a Kalman smoothing algorithm [34]. The instrumented split-belt treadmill recorded the ground reaction forces under each foot separately at a frequency of 1000 Hz. From the kinematics and ground reaction forces, the timing of the initial contact (IC), toe off (TO), foot flat (FF) and heel off (HO) event was calculated, following the conventions from [35]: IC occurs when the vertical force exceeds 20 N; TO occurs when the vertical force drops below 20 N; FF is the first of at least ten time instants after IC which have a vertical toe velocity lower than 100 mm/s; HO is the last of at least ten time instants before TO that have a vertical heel velocity lower than 100 mm/s.

IV. Method

Fig. 2 shows the outline of the method. In the offline learning phase, see Section IV-A, a probabilistic model for joint trajectories and gait events of a leg during one gait cycle is learned from the repetitive demonstrations of human walking patterns. The result is a probabilistic model in function of a normalized time s, indicating the progress along the trajectory,

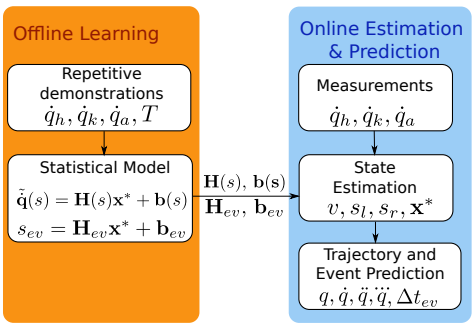

Fig. 2. The outline of the method. In the offline learning phase, a probabilistic model for the joint trajectories and gait events is learned from the dataset. Online, we first estimate the state from the available measurements. With this state we can predict future joint trajectories, their derivatives and gait events.

and some latent variables $\mathbf{x}^*$. In the online estimation phase, see Section IV-B, joint velocity measurements are used to provide an estimate for the latent variables $\mathbf{x}^*$ together with an estimate of the current normalized time of left and right leg and an estimate of the stride time. Once we know the complete state, we can predict future joint trajectories, their derivatives and gait events for both sides, see Section IV-C.

In contrast to machine learning approaches that use a short-term history of the position or velocity signals as input such as [19], [20], our approach has the advantage that it can take into account a longer-term history. In contrast to the approach of e.g. Jezernik [12], our approach can handle more than scaling in amplitude and time or adding an offset: during the offline learning phase, our approach will determine the shape of "modes" by which the trajectories can vary; during the online phase, we estimate the optimal amplitude of these modes in order to predict future events and the future evolution of the trajectories. The scaling in time is handled explicitly during the online phase.

A. Offline learning phase

In the offline learning phase, a gait dataset is used to learn joint trajectories and gait events following the ideas of [29]. A model from a dataset of joint trajectories is constructed by applying probabilistic principal component analysis (PPCA). The PPCA is an adaptation of principal component analysis (PCA) that is used to describe a high-dimensional Gaussian probability distribution using a model that only involves a limited number of variables [36]. Similarly to PCA, it is a dimensionality reduction technique, but it is applied to probability distributions. A procedure to compute the PPCA is readily available in Matlab [37]. The offline learning approach explained below samples the joint trajectories into a larger number of discrete points; constructs a covariance matrix to represent the distribution of the trajectories; and reduces the dimensionality of this probabilistic model using PPCA. Afterwards, a probabilistic model that is a continuous function of normalized time s is obtained by fitting the discretized model. The result is a probabilistic model that concisely describes the probabilistic variations of the joint trajectories and gait events.

As joint angles can be affected by offsets, we choose to model joint angular velocities. The trajectories are modeled in function of a normalized time $s = t/T$, with t time and T the stride time. Under the assumption that the amplitudes of the joint angle trajectory are approximately independent of walking speed, we define normalized joint velocities $\tilde{\dot{q}}$ as:

$$\tilde{\dot{q}} = T\dot{q}. \tag{1}$$

A vector function $\mathbf{y}(s)$ represents all the information of a stride (from IC at $s = 0$ to IC at $s = 1$):

$$\mathbf{y}(s) = \begin{bmatrix} \tilde{\dot{q}}_h(s) & \tilde{\dot{q}}_k(s) & \tilde{\dot{q}}_a(s) & s_{FF} & s_{HO} & s_{TO} \end{bmatrix}^T, \tag{2}$$

where the indices h, k and a respectively refer to the hip, knee and ankle joint, whereas s_{FF}, s_{HO} and s_{TO} refer to the normalized timing of FF, HO and TO. The timing of IC is implicitly represented by our method since, by construction,

IC always occurs at s = 0, 1, 2, 3, ... where each integer value marks the start of the gait cycle of a leg.

Using a collected gait dataset, we construct a compact probabilistic model for $\mathbf{y}(s)$ by first discretizing this vector function into a discrete vector $\mathbf{y}_s$. The velocity of the hip, knee and ankle are sampled at d equidistant normalized times s_1, s_2, ..., s_d with $s_1 = 0$ and $s_d = 1$, to obtain a discrete vector $\mathbf{y}_s \in \mathbb{R}^{3d+3}$:

$$\begin{aligned}\mathbf{y}_s^T = [&\tilde{q}_h(s_1) \dots \tilde{q}_h(s_d) \quad \tilde{q}_k(s_1) \dots \tilde{q}_k(s_d) \\ &\tilde{q}_a(s_1) \dots \tilde{q}_a(s_d) \quad s_{FF} \quad s_{HO} \quad s_{TO}],\end{aligned} \tag{3}$$

where d is the number of discrete normalized sample times. Before constructing the model, the vector $\mathbf{y}_s^T$ is weighted with a weighting matrix $\mathbf{W}_s$:

$$\mathbf{y}_{s,w} = \mathbf{W}_s \cdot \mathbf{y}_s \tag{4}$$

with

$$\mathbf{W}_s = \text{diag}(\mathbf{I}_{3d}, w_{FF}, w_{HO}, w_{TO}), \tag{5}$$

where $\mathbf{I}_{3d}$ represents a unit matrix with size $3d$. This weight matrix $\mathbf{W}_s \in \mathbb{R}^{(3d+3)\times(3d+3)}$ makes a trade-off between the $3d$ columns representing trajectory velocities and the three columns representing the timing of each event. Higher weights w_{FF}, w_{HO}, w_{TO} will result in better modeling of the events and worse modeling of the trajectories. If some joints have less accurate measurements, we can choose the weights inversely proportional to the covariance of the measurement uncertainty. Here, we assume all joints to have the same accuracy.

In our learning approach, we construct $\mathbf{y}_{s,w}^i$ for all strides i in the dataset. We assume a symmetric gait pattern, so we consider the left and right side as separate samples of the same distribution. The methodology of this paper is however still applicable without this assumption, but more data will be necessary for the same performance.

The PPCA procedure [36] [37] is used to implicitly construct a covariance matrix and to obtain a reduced model for the probability distribution of $\mathbf{y}_{s,w}$. This reduced model for $\mathbf{y}_{s,w}$ is described by m modes in a matrix $\mathbf{H}_{s,w} \in \mathbb{R}^{(3d+3)\times m}$ (each corresponding to a column), a mean $\mathbf{b}_{s,w} \in \mathbb{R}^{3d+3}$, and a remaining uncertainty $\boldsymbol{\varepsilon}_{s,w}$ in the following way:

$$\mathbf{y}_{s,w} = \mathbf{H}_{s,w}\mathbf{x}^* + \mathbf{b}_{s,w} + \boldsymbol{\varepsilon}_{s,w}, \tag{6}$$

with $\boldsymbol{\varepsilon}_{s,w} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_{3d+3})$. The values of $\mathbf{H}_{s,w}$, $\mathbf{b}_{s,w}$ and σ are determined using PPCA [36], [37]. Using (4), $\mathbf{y}_s$ can be written as:

$$\mathbf{y}_s = \mathbf{H}_s\mathbf{x}^* + \mathbf{b}_s + \boldsymbol{\varepsilon}_s, \tag{7}$$

with $\mathbf{H}_s = \mathbf{W}_s^{-1}\mathbf{H}_{s,w}$, $\mathbf{b}_s = \mathbf{W}_s^{-1}\mathbf{b}_{s,w}$ and $\boldsymbol{\varepsilon}_s \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{W}_s^{-1}\mathbf{W}_s^{-T}\right)$. Using this model, each instance of $\mathbf{y}_s \in \mathbb{R}^{3d+3}$ is described by a reduced vector $\mathbf{x}^* \in \mathbb{R}^m$, which has a normal distribution $\mathbf{x}^* \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{mm})$. $\mathbf{x}^*$ represents the prior knowledge on the trajectories and events. In the online learning phase we will introduce additional knowledge and measurements to provide improved estimates for $\mathbf{x}^*$.

In (7), the elements of $\mathbf{H}_s$ and $\mathbf{b}_s$ can be grouped as follows:

$$\mathbf{y}_s = \begin{bmatrix}\mathbf{H}_h(s_1 \dots s_d) \\ \mathbf{H}_k(s_1 \dots s_d) \\ \mathbf{H}_a(s_1 \dots s_d) \\ \mathbf{H}_{FF} \\ \mathbf{H}_{HO} \\ \mathbf{H}_{TO}\end{bmatrix}\mathbf{x}^* + \begin{bmatrix}\mathbf{b}_h(s_1 \dots s_d) \\ \mathbf{b}_k(s_1 \dots s_d) \\ \mathbf{b}_a(s_1 \dots s_d) \\ \mathbf{b}_{FF} \\ \mathbf{b}_{HO} \\ \mathbf{b}_{TO}\end{bmatrix} + \boldsymbol{\varepsilon}_s, \tag{8}$$

$$\mathbf{H}_j(s_1 \dots s_d) = \begin{bmatrix}H_{j,1}(s_1) & \dots & H_{j,m}(s_1) \\ \vdots & \ddots & \vdots \\ H_{j,1}(s_d) & \dots & H_{j,m}(s_d)\end{bmatrix}, \tag{9}$$

$$\mathbf{b}_j(s_1 \dots s_d) = [b_j(s_1) \quad b_j(s_2) \dots b_j(s_d)]^T, \tag{10}$$

with $j = h$ for the hip, k for the knee, and a for the ankle, and $s_1 \dots s_d$ the normalized discrete times. Equation (8) models the joint velocity trajectories at discrete normalized time instances. To model the continuous trajectories, we fit a Fourier Series (FS) of the form

$$\text{FS} = \sum_k a_{c,k}\cos(2\pi k s) + \sum_k a_{s,k}\sin(2\pi k s) \tag{11}$$

through each column of $\mathbf{H}_j$ and through $\mathbf{b}_j$. The Fourier series do not have an offset parameter, as the mean joint velocity over a gait cycle should be zero. The period of the series is equal to one, as the normalized time also has a period of one. Using these Fourier series, we obtain a probabilistic model for the trajectories and the events, that is continuous in normalized time s, and has only a limited number of remaining variables $\mathbf{x}^*$:

$$\mathbf{y}(s) = \begin{bmatrix}\mathbf{H}_h(s) \\ \mathbf{H}_k(s) \\ \mathbf{H}_a(s) \\ \mathbf{H}_{FF} \\ \mathbf{H}_{HO} \\ \mathbf{H}_{TO}\end{bmatrix}\mathbf{x}^* + \begin{bmatrix}b_h(s) \\ b_k(s) \\ b_a(s) \\ b_{FF} \\ b_{HO} \\ b_{TO}\end{bmatrix} + \boldsymbol{\varepsilon} \tag{12}$$

$$= \mathbf{H}(s)\,\mathbf{x}^* + \mathbf{b}(s) + \boldsymbol{\varepsilon}, \tag{13}$$

$$\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{W}^{-1}\mathbf{W}^{-T}), \tag{14}$$

$$\mathbf{W} = \text{diag}(\mathbf{I}_3, w_{FF}, w_{HO}, w_{TO}). \tag{15}$$

This model, periodic in s with a period equal to 1, remains valid for $s \geq 1$ and can be interpreted in the following way: a joint trajectory or gait event time is described by the mean joint trajectory or event time, represented by the 6×1 vector of Fourier functions $\mathbf{b}(s)$, plus deviations from the mean, represented by $\mathbf{H}(s)\mathbf{x}^*$. The model correlates the deviations of one joint to the deviations of another joint and from one time instant to another. The columns of the $6 \times m$ matrix $\mathbf{H}(s)$ can be seen as m basis functions or modes describing these coupled deviations from the mean. The latent variables $\mathbf{x}^*$ are the coefficients of these basis functions. These columns represent the principal components, i.e., the directions of the largest variability in the dataset. The number of basis functions m is chosen at the start of PPCA. It should be high enough to have a good model, but not too high to avoid overfitting. However, m is also limited by the computational load during real-time execution.

B. Online estimation phase

Using the methods of subsection IV-A, we can concisely describe the joint trajectories and gait events belonging to a single side of the body using a state vector $\mathbf{x}^*$. Under the assumption of a symmetric gait pattern, we can extend this to both sides of the body using a state $\mathbf{x} \in \mathbb{R}^{m+3}$:

$$\mathbf{x} = [v, s_l, s_r, \mathbf{x}^{*T}]^T, \tag{16}$$

$$\text{lb}_{\Delta s} \leq s_l - s_r \leq \text{ub}_{\Delta s}, \tag{17}$$

where v is the reciprocal of the stride time $T : v = T^{-1}$, and $\mathbf{x}^*$ describes the gait pattern of both left and right side, and s_l the normalized time for the left side and s_r for the right side. The difference between s_l and s_r is constrained to be approximately 0.5 (half a gait cycle) with an upper and lower bound.

An estimator estimates this state from the joint velocity measurements of both left and right side (see Fig. 2). We follow a Bayesian approach starting from a prior belief in the state $\mathbf{x}$, represented by a mean $\bar{\mathbf{x}}_0$ and covariance matrix $\mathbf{P}_0$. We model how this belief evolves over time using a *process model*. Measurements, described by a *measurement model* are used to update this belief, resulting in a posterior belief, and this procedure is repeated over time. The (iterative extended) Kalman filter [29] is such an approach. However, a Kalman filter can suffer from robustness problems due to the strong nonlinearity of process and measurement equations. Therefore, we choose a *moving horizon estimator* (MHE) to handle this estimation problem. The MHE takes into account all the measurements inside a sliding time window by maximizing the posterior probability to estimate the (not directly observable) state $\mathbf{x}_i$ [38]:

$$\min_{\mathbf{x}_\cdot, \mathbf{e}_{p,\cdot}, \mathbf{e}_{m,\cdot}, v_\cdot} \quad \boldsymbol{\mu}_0^T \mathbf{P}_0^{-1} \boldsymbol{\mu}_0 + \sum_{i=0}^{N-1} \mathbf{e}_{p,i}^T \mathbf{Q}^{-1} \mathbf{e}_{p,i} + \sum_{i=0}^{N-1} \mathbf{e}_{m,i}^T \mathbf{R}_i^{-1} \mathbf{e}_{m,i} \tag{18}$$

subject to:

$$\mathbf{x}_{i+1} = \mathbf{F}\mathbf{x}_i + \mathbf{e}_{p,i} \qquad \forall i = 0 \dots N-1 \tag{19}$$

$$\mathbf{z}_i = \mathbf{h}(\mathbf{x}_{i+1}) + \mathbf{e}_{m,i} \quad \forall i = 0 \dots N-1 \tag{20}$$

$$\boldsymbol{\mu}_0 = \mathbf{x}_0 - \bar{\mathbf{x}}_0 \tag{21}$$

$$\text{lb}_{\mathbf{x}^*} \leq \quad \mathbf{x}_i^* \quad \leq \text{ub}_{\mathbf{x}^*} \qquad \forall i = 0 \dots N \tag{22}$$

$$\text{lb}_v \leq \quad v_i \quad \leq \text{ub}_v \qquad \forall i = 0 \dots N \tag{23}$$

$$\text{lb}_{\Delta s} \leq \; s_{l,i} - s_{r,i} \; \leq \text{ub}_{\Delta s} \quad \forall i = 0 \dots N \tag{24}$$

This optimization problem is solved iteratively using a window size N. $N = 1$ is equivalent to an extended (i.e., linearized) Kalman filter. The variables and equations in this optimization problem are explained in the next paragraphs.

The process model (19) describes the probabilistic evolution of the state. We assume that the stride time T, hence also v, and the latent variables $\mathbf{x}^*$ are approximately constant over time and that the dimensionless time s_l and s_r vary linearly with time:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 0 & \mathbf{O}_{1\times m} \\ \Delta t & 1 & 0 & \mathbf{O}_{1\times m} \\ \Delta t & 0 & 1 & \mathbf{O}_{1\times m} \\ \mathbf{0}_{m\times 1} & \mathbf{0}_{m\times 1} & \mathbf{0}_{m\times 1} & \mathbf{I}_m \end{bmatrix}. \tag{25}$$

The process noise $\mathbf{e}_{p,i}$ describes how much $\mathbf{x}^*$, v, s_l and s_r can deviate from their nominal behavior; its covariance matrix is described by a diagonal matrix:

$$\mathbf{Q} = \text{diag}(Q_v, Q_{s_l}, Q_{s_r}, \mathbf{Q}_x). \tag{26}$$

The optimization criterion (18) contains a term that minimizes this process uncertainty $\mathbf{e}_{p,i}$, weighted with $\mathbf{Q}^{-1}$.

The measurement model (20) expresses how well the measurements $\mathbf{z}_i$ are explained by the current state $\mathbf{x}_i$. The measurements $\mathbf{z}_i$ consist of joint velocities:

$$\mathbf{z}_t = [\dot{q}_{h_l}(t) \;\; \dot{q}_{k_l}(t) \;\; \dot{q}_{a_l}(t) \;\; \dot{q}_{h_r}(t) \;\; \dot{q}_{k_r}(t) \;\; \dot{q}_{a_r}(t)]^T. \tag{27}$$

Combining (1),(2) and (13) leads to a probabilistic relationship between the state and these velocities:

$$\dot{q}_j(t) = \frac{1}{T}\Big(\mathbf{H}_j(s)\mathbf{x}^* + b_j(s)\Big) + \delta_j, \tag{28}$$

where j= h, k, or a. $\delta_j \sim \mathcal{N}(\mathbf{0}, \frac{\sigma^2}{T^2}\mathbf{I})$ represents the remaining measurement uncertainty, which can be deduced from the learned model in section IV-A, see (1), (2), and (12)–(15). This leads to an expression for $\mathbf{h}(\mathbf{x}_i)$:

$$\mathbf{h}(\mathbf{x}_i) = v_t \left(\begin{bmatrix} \mathbf{H}_h(s_{l,i}) \\ \mathbf{H}_k(s_{l,i}) \\ \mathbf{H}_a(s_{l,i}) \\ \mathbf{H}_h(s_{r,i}) \\ \mathbf{H}_k(s_{r,i}) \\ \mathbf{H}_a(s_{r,i}) \end{bmatrix} \mathbf{x}_t^* + \begin{bmatrix} b_h(s_{l,i}) \\ b_k(s_{l,i}) \\ b_a(s_{l,i}) \\ b_h(s_{r,i}) \\ b_k(s_{r,i}) \\ b_a(s_{r,i}) \end{bmatrix} \right), \tag{29}$$

and for the variance of the measurement uncertainty:

$$\mathbf{R}_i = v_t^2 \sigma^2 \mathbf{I}_6. \tag{30}$$

The matrix $\mathbf{W}$ does not appear in this equation, since the weights belonging to the joint velocities are equal to one. This measurement equation is nonlinear in s. The optimization criterion (18) contains a term that minimizes the measurement uncertainties $\mathbf{e}_{m,i}$, weighted with $\mathbf{R}_i^{-1}$.

The term $\boldsymbol{\mu}_0^T \mathbf{P}_0^{-1} \boldsymbol{\mu}_0$ of the criterion (18) corresponds to our prior belief for the state $\mathbf{x}$, described by a mean $\bar{\mathbf{x}}_0$ and variance $\mathbf{P}_0$. This belief is based on older measurements, outside of the optimization window. The result of the optimization will be used to update $\mathbf{P}_0$ and $\bar{\mathbf{x}}_0$ for the next instance of the sliding window. $\mathbf{P}_0$ has to take into account that a single measurement is part of many sliding windows, to avoid counting the same information multiple times. See [39] for a complete description of these updates.

The last three constraints (22)–(24) represent bounds on $\mathbf{x}^*$ and v and on the normalized time difference of the left and right leg (see (17)).

C. Online prediction phase

In the prediction phase, we use the estimation of the current state to predict joint velocities at a future instant. Using the process equation (19) we can evaluate the state at a future time t:

$$v(t) = v_{now} \tag{31}$$
$$s_l(t) = s_{l,now} + v_{now}(t - t_{now}) \tag{32}$$
$$s_r(t) = s_{r,now} + v_{now}(t - t_{now}) \tag{33}$$
$$\mathbf{x}^*(t) = \mathbf{x}^*_{now} \tag{34}$$

With this state at time t, we can evaluate the measurement equation (29) to get the joint velocities at t. Recall that b_h, b_k, b_a and each column of $\mathbf{H}_h$, $\mathbf{H}_k$, $\mathbf{H}_a$ are written as a Fourier series. Substituting (32) and (33) in (11) results in analytical functions in t. Therefore, equation (29) can be integrated and differentiated analytically w.r.t. t to get the joint angles, accelerations, and higher order derivatives. The integration constant can be determined from the current angle measurements.

The normalized time of a gait event is computed as:

$$s_{ev} = w_{ev}^{-1}\left(\mathbf{H}_{ev}\mathbf{x}^* + b_{ev}\right) \tag{35}$$

with ev = FF, HO or TO. By convention, $s_{ev} = 0$ for IC. The time until the left side event ($s = s_l$) or the right ($s = s_r$) can be computed as:

$$\Delta t_{ev} = \begin{cases} \dfrac{s_{ev} - mod(s,1)}{v} & s_{ev} \geq mod(s,1) \\ \dfrac{s_{ev} + 1 - mod(s,1)}{v} & s_{ev} < mod(s,1) \end{cases} \tag{36}$$

with mod the modulo operator. If $s_{ev} < mod(s,1)$, then the event already occurred in the current gait cycle. Hence we add one to calculate the time to the event in the next gait cycle.

V. Validation experiments and results

In section III, two datasets were collected, dataset 1 contains data on walking without an exoskeleton, dataset 2 data on walking with an exoskeleton. Fig. 3 shows the timing of the measured gait events for these two datasets. Walking with an exoskeleton results in slightly different walking patterns. For trials with exoskeleton, the stance phase of a leg is longer than for trials without exoskeleton, since the TO event appears later in the gait cycle.

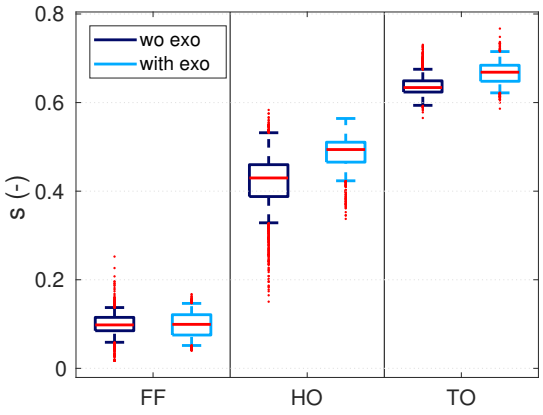

Fig. 3. The timing of the measured gait events. The IC event by definition occurs at $s = 0$ in both cases. The horizontal red line in the boxplot marks the median. The box ranges from the 25th to the 75th percentile. The whiskers contain 90% of the data.

Table I describes the two validation experiments that are performed using these collected datasets. In a first validation experiment (*without exoskeleton*), a cross-validation approach is used where dataset 1 is split into a validation set, consisting of all gait cycles of one subject, and a learning dataset, consisting of all gait cycles of the other subjects. This is repeated for every subject in the dataset until each subject is used for validation and excluded for learning once. In a second validation experiment (*with exoskeleton*), the learning data set is the full dataset 1, the validation dataset is full dataset 2. Note that the second validation experiment, with exoskeleton, is very strong: we extrapolate from walking without an exoskeleton to a situation were the walking pattern is disturbed by an exoskeleton.

The learning phase consists of the methods explained in subsection IV-A, the validation phase consists of the methods explained in subsections IV-B and IV-C. These methodologies require a number of hyperparameters that are described in Table II. Window size N, number of modes m and sample time Δt are determined by the trade-off between computation time and accuracy. We assumed a reasonable requirement for the sample time $\Delta t = 0.01s$, and determined the largest achievable m and N using both datasets. The number of discretization points d for the learned model has to be chosen sufficiently high and does not influence online computation time, since we constructed a continuous model in the offline

TABLE I
DESCRIPTION OF THE TWO VALIDATION EXPERIMENTS

Experiment name	Validation approach
Without exoskeleton	
Learning	dataset 1 (leave 1 subject out)
Validation	dataset 1 (28-fold cross-validation)
With exoskeleton	
Learning	dataset 1
Validation	dataset 2

TABLE II
VALUE OF THE USED HYPERPARAMETERS.

Hyperparameter	Value	Hyperparameter	Value
d	50	Δt	0.01 s
m	10	N	10
w_{FF}	50	$\text{lb}_{\mathbf{x}^*}$	-5
w_{HO}	50	$\text{ub}_{\mathbf{x}^*}$	5
w_{TO}	50	lb_v	$0.4s^{-1}$
Q_v	$1e-6s^{-2}$	ub_v	$1.5s^{-1}$
Q_{s_l}	$1e-6$	$\text{lb}_{\Delta s}$	0.45
Q_{s_r}	$1e-6$	$\text{ub}_{\Delta s}$	0.55
$\mathbf{Q_x}$	$1e-6 \cdot \mathbf{I}$		

learning phase. The event weights w_{FF}, w_{HO} and w_{TO} are chosen such that all event variables and all trajectory variables in the vector $\mathbf{y}_s$ are weighed equally. The bounds lb_v and ub_v on the inverse stride time v are determined using both datasets, and are chosen such that the system does not extrapolate too much outside the range that we verified in these validation experiments. The bounds $\text{lb}_{\mathbf{x}^*}$ and $\text{ub}_{\mathbf{x}^*}$ on $\mathbf{x}^*$ are applied for the same reason, and their value is not critical for the results. With the bounds $\text{lb}_{\Delta s}$ and $\text{ub}_{\Delta s}$ on the difference between the progress left s_l and right s_r, we enforce our knowledge that the subjects walk with an approximately symmetrical gait pattern.

The process noise hyperparameters Q_v, Q_{s_l}, Q_{s_r} and $\mathbf{Q}_{\mathbf{x}}$ determine how variable v, s_l, s_r and $\mathbf{x}^*$ are during walking. For an ideal, systematic walking pattern, they should remain constant; in reality there are variations in the human walking pattern. The lower we set these process hyperparameters, the more the estimator averages out the measurements and the higher the estimation robustness of the gait pattern, but the less the system can describe the variations in the gait pattern. For these validation experiments we determined (manually, by order of magnitude) reasonable values based on dataset 1 and 2.

Fig. 4 illustrates the joint trajectory prediction for walking without an exoskeleton. At each time instant t, joint angle, velocity, acceleration and jerk are predicted for time points A, B and C relative to t. Point A, B and C are located in the future, respectively 10%, 50% and 100% of a gait cycle. Fig. 4 shows the estimation in progress. Initially, with no measurements available, the prediction is equal to the mean of the dataset. The shape of the prediction is similar to the measured pattern, but the amplitude and timing need improvement. After only a few measurements the timing and amplitude estimation improves. After still more observations, the amplitude is estimated even better.

Fig. 5 shows the prediction of the time to IC without exoskeleton. The predicted time to IC follows the true time to IC very closely except during the first step where the estimator still needs to converge.

Table III reports the accuracy of the trajectory prediction and the estimation of higher order derivatives: the table shows the mean and standard deviation of the trajectory estimation errors of q, $\dot{q}$, $\ddot{q}$, $\dddot{q}$ for points A,B, and C, and for hip, knee and ankle joint, for trials with and without exoskeleton. Predicted and measured values at the three points are compared in order to validate the joint trajectory prediction in short, mid and long term. The error between predicted and measured signals are reported as a percentage of the range of the signal during a trial.

Fig. 6 validates the event prediction: we calculate the predicted time until the next event at 0.2 seconds before the true event. Hence, the prediction error is zero if the predicted time to the event is equal to 0.2. If the predicted time is larger than 0.2, the event is predicted later than the true event.

Lastly, Fig. 7 compares the result of our proposed methodology with a simpler method using only the mean of the trajectories and the event timings (i.e., $m = 0$). The effect of learning a probabilistic model and using latent variables on the standard deviation of the gait trajectory predictions and the interquartile range of the event predictions is shown. For the case without the latent variables, the process noise was lowered with a factor 10 to improve robustness.

VI. Discussion

Some general trends are visible in Table III. All predictions are unbiased since the mean error is in all cases very close to zero. The standard deviation of the estimations show differences. So, although the estimates are on average very close to their ground truth value, individual estimates can vary with a standard deviation ranging from 4.4% for the knee joint angle at a prediction horizon of 10% without the exoskeleton, to 28.5% for the ankle joint position at a prediction horizon of 50%, with the exoskeleton. For the trials without exoskeleton, higher order derivatives systematically have a higher standard deviation of the error; for the trials with exoskeleton there is no similar systematic trend. For the first, second and third derivatives, the prediction gets progressively worse for points B and C further in the future. For the joint angle positions at time horizon C (100%), there is no such trend. Furthermore, except for the hip joint angle at time horizon C (100%) with and without exoskeleton, and the third derivative of the ankle angle with exoskeleton, the knee is better predicted than the hip or ankle joint. This can be explained by the knee velocity trajectory that has three well distinct extrema during one gait cycle, around $t = 0.2s$, $t = 0.8s$ and $t = 1.1s$, see Fig. 4. Measurements of these extrema gives information on both the timing and amplitude of the trajectory. Our method can take this implicitly into account, resulting in a better estimation. In all cases, the standard deviation of joint angle prediction errors (and their higher order derivatives) is lower in trials without exoskeleton than in trials with exoskeleton. This can be expected because the training data is only based on trials without exoskeleton, and because there are differences between the trials with and without an exoskeleton, see Fig. 3.

Fig. 6 shows the error statistics of the event prediction at 0.2 seconds before the true event. For the dataset without exoskeleton, the (absolute value of the) median error is lower than 9 ms for all events. Also, the contours of the box containing the 50% lowest errors are very close to zero. The HO events are the most difficult to predict, as can be seen from the interquartile range and the 90% bounds. Compared to the other events, the measured HO events vary more in the dataset, see Fig. 3. Because the model was only trained on a dataset for walking without exoskeleton, the prediction error with exoskeleton is worse than without exoskeleton. For the case with exoskeleton, the median of the prediction errors for the IC and TO events are 15 and 33ms respectively. In the control of the exoskeleton, this bias could be subtracted from the prediction. However, these statistics are only based on 5 subjects all walking in the same exoskeleton. This is not enough to have good estimates of the bias. Remarkably, the prediction error of HO is on average only 19 ms, although Fig. 3 shows that HO with exoskeleton occurs significantly later in the gait cycle compared with the event without exoskeleton. However, the HO event remains hard to predict,

TABLE III
MEAN AND STANDARD DEVIATION (BETWEEN PARENTHESES) OF THE JOINT ANGLE, VELOCITY, ACCELERATION AND JERK PREDICTION ERRORS FOR POINTS AT 10% (A), 50% (B) AND 100% (C) OF A GAIT CYCLE IN THE FUTURE. THE ERROR IS EXPRESSED AS A PERCENTAGE RELATIVE TO THE RANGE OF THE SIGNAL DURING A TRIAL.

		Without Exoskeleton			With Exoskeleton		
		A	B	C	A	B	C
hip	q	-0.1% (4.9%)	-0.5% (9.1%)	-0.0% (7.4%)	-0.2% (9.2%)	-0.6% (14.7%)	-0.6% (10.6%)
	$\dot{q}$	-0.3% (7.0%)	0.3% (8.0%)	0.3% (9.0%)	-0.3% (12.2%)	0.1% (13.6%)	-0.3% (14.3%)
	$\ddot{q}$	-0.2% (9.8%)	-0.1% (10.9%)	-0.4% (12.1%)	0.1% (14.9%)	-0.1% (15.6%)	-0.3% (16.9%)
	$\dddot{q}$	0.2% (12.1%)	-0.0% (12.7%)	-0.3% (13.7%)	0.1% (15.1%)	0.2% (15.2%)	-0.2% (15.9%)
knee	q	0.3% (4.4%)	-0.1% (8.7%)	-0.1% (8.8%)	0.2% (7.2%)	0.0% (13.1%)	0.6% (12.0%)
	$\dot{q}$	0.4% (4.5%)	-0.1% (6.7%)	-0.1% (8.5%)	0.4% (8.0%)	-0.4% (10.5%)	-0.7% (11.8%)
	$\ddot{q}$	0.0% (6.3%)	0.4% (8.7%)	0.7% (10.5%)	-0.0% (10.8%)	0.4% (13.7%)	0.4% (14.8%)
	$\dddot{q}$	-0.3% (8.4%)	-0.0% (9.7%)	0.2% (11.2%)	-0.3% (12.6%)	-0.1% (14.0%)	0.3% (14.4%)
ankle	q	0.4% (8.8%)	0.1% (15.1%)	-0.2% (9.4%)	1.6% (19.2%)	1.0% (28.5%)	-0.4% (14.5%)
	$\dot{q}$	0.4% (7.8%)	0.1% (9.3%)	0.1% (10.7%)	0.6% (13.3%)	0.0% (14.0%)	0.2% (14.9%)
	$\ddot{q}$	-0.1% (8.8%)	0.0% (10.7%)	0.2% (12.5%)	-0.4% (13.2%)	-0.0% (14.2%)	0.7% (15.7%)
	$\dddot{q}$	-0.3% (9.7%)	-0.0% (11.6%)	0.2% (13.3%)	-0.0% (13.1%)	-0.0% (13.6%)	0.1% (14.5%)

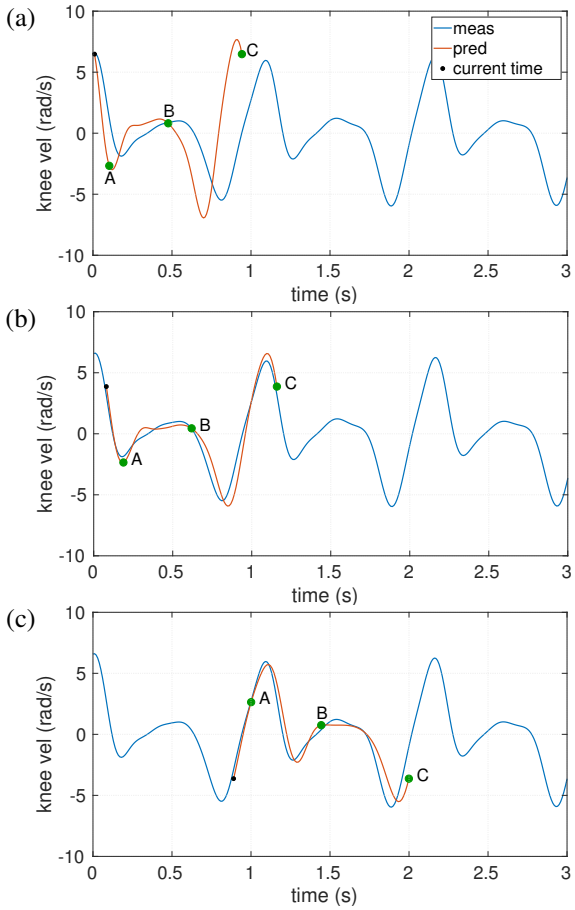

Fig. 4. The measured (blue) and predicted (red) knee velocity without exoskeleton. The prediction gets better when more measurements become available. The errors between measured and predicted signal are evaluated at points A, B and C.

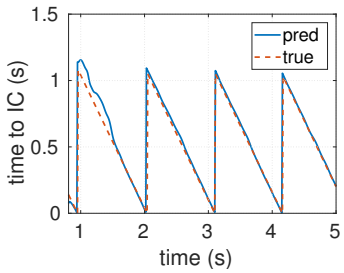

Fig. 5. The predicted time to IC (blue) and true time to IC (red) for trials without exoskeleton.

as it has a large interquartile range and 56 outliers (out of 689 analyzed time instances) bigger than 1 s. These outliers occur because the prediction using the decision rule (36) in those cases assumes that HO already occurred at 0.2 s before the true event and is already predicting the next HO event. Other decision policies can be designed, depending on the controller application. A controller can use the prediction of IC, FF and TO to generate a smooth transition from one control mode to another.

The current method is suitable for real-time computation. On average the estimation and prediction phase took 6 ms per time instant on a laptop with an Intel Core i7-4800MQ 2.7 GHz processor, which is fast enough to update the predictions every 0.01 s as indicated in Table II.

Fig. 7 shows that, for the validation without exoskeleton, using the latent variables is certainly better than just using the population mean: the spread on trajectory and gait event predictions decreases with 5 to 30%. With the exoskeleton, using the latent variables is better for predicting joint trajectories except for predicting the jerks where the prediction error does not improve. Furthermore, the interquartile range on FF event decreases with 30%, but the interquartile ranges of the other events increases 30% to 50% compared to the prediction with only the population mean. This shows that the probabilistic model is able to capture the correlations between the kinematics and the events for the validation without exoskeleton.

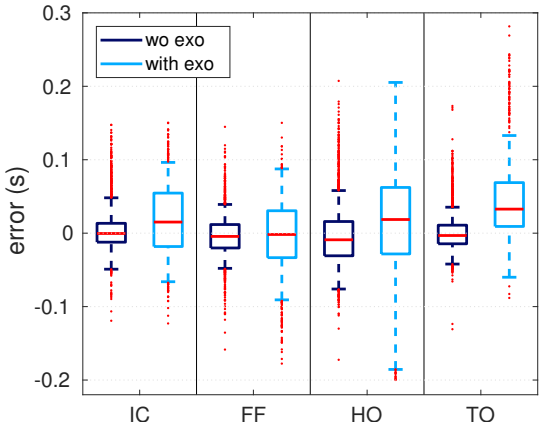

Fig. 6. The errors of gait event prediction using latent variables for trials with and without exoskeleton at 0.2 s before the true event. The horizontal red line in the boxplot marks the median. The box ranges from the 25th to the 75th percentile. The whiskers contain 90% of the data. The HO prediction with exoskeleton has 56 outliers bigger than 1 s which are not shown for readability of the graph.

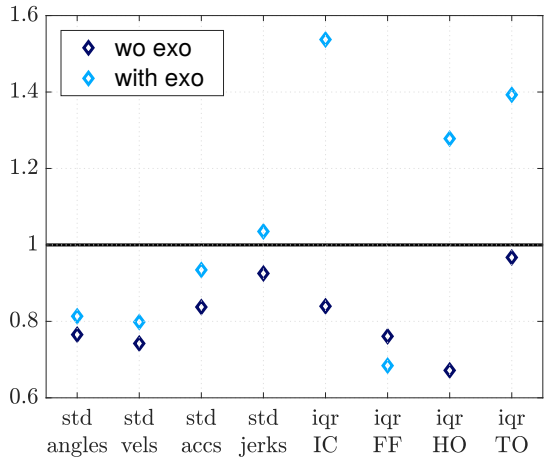

Fig. 7. Improvement of using the latent variables on the standard deviation and interquartile range of the prediction errors for the joint and event prediction. The result with latent variables is reported relative to the result with only using the mean trajectories and timing events. Values above 1 indicate that the result with latent variables is larger than the result with just using the mean.

These correlations are captured inside the $\mathbf{H}$ matrix. The latent variables are estimated solely from observation of the joint velocities without event observations. However, this learned correlation is not sufficient to extrapolate the model to the validation with exoskeleton, leading to the worse predictions of the event timings in Fig. 7. This can be solved in two ways: i) include exoskeleton trials in the learning phase, ii) add exoskeleton sensors that reliably detect gait events, e.g., IC event can be measured with a foot switch or a distance sensor.

The probabilistic model can easily include such additional information, e.g., when the left IC event is detected, an additional measurement equation is added to express that the normalized time is approximately equal to the closest integer multiple of 1, i.e. $s_l \mod 1 \approx 0$. . Similarly, we can use (2) and (12)–(15) to establish a measurement model for the other events. In this way, latent variables can be estimated based on both kinematics and event information.

The proposed probabilistic methodology also includes bounds on the latent variables $\mathbf{x}^*$ indicating that the shape of the trajectories should not change to drastically, and bounds on $v = T^{-1}$, indicating that the stride time should only be allowed to vary within a given range. The bounds on Δs indicate the expected symmetry of the gait pattern. A big advantage of our approach is that we can determine how far our model should extrapolate outside the range of $\mathbf{x}^*$, v and Δs encountered during training. The bounds have a clear physical meaning, and are therefore easier to tune, compared to black-box machine-learning techniques.

In future work, results can be improved by learning from a dataset that includes exoskeleton trials. For validation, this will however require more subjects than the five that were available for this study. An exoskeleton disturbs the walking pattern. This disturbance depends on the ability of the exoskeleton controller to actively compensate its own dynamics and friction. The signals generated by the gait prediction methodology of this paper could significantly improve the performance of such controller by generating an appropriate feedforward (partially) compensating dynamics and friction [27]. For healthy gait, we therefore envision the following procedure: first, learn a model solely based on trials without exoskeleton. Use this model to (partially) improve the performance of the exoskeleton controller and acquire a new training dataset. Then repeat this procedure until there are no further improvements.

The computed probabilities could also be used inside the control strategy, e.g. by controlling the joints of an exoskeleton using an impedance matrix that is proportional to the inverse of the covariance matrix at a point on the probabilistic trajectory model, in a similar way is as done in [28] for the control of a robot arm. Another example would be an exoskeleton controller that checks whether measurements are still conforming to the probabilistic model, and use this to decide when to switch control modes.

VII. Conclusion

In this work we presented an integrated, real-time probabilistic methodology to estimate a single, consistent model to estimate higher order derivatives of the joint angles and to predict future gait joint trajectories and events from measured kinematics.

The method was trained on a dataset for walking without exoskeleton; afterwards the method was validated: Using cross-validation, we showed that our method is effective for walking without exoskeleton for both trajectory and event prediction. Since the walking pattern for walking with an exoskeleton is disturbed, the dataset for walking with an exoskeleton represented an extrapolation of the trained model. The gait trajectory prediction was still improved, but the results of event prediction were mixed.

The ability to predict future joint trajectories and gait events offers opportunities to design exoskeleton controllers which anticipate these trajectories and events, allowing better tracking control and smoother, accurately timed transitions between different control modes.

REFERENCES

[1] Jan F Veneman, Rik Kruidhof, Edsko EG Hekman, Ralf Ekkelenkamp, Edwin HF Van Asseldonk, and Herman Van Der Kooij. Design and evaluation of the lopes exoskeleton robot for interactive gait rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(3):379–386, 2007.

[2] Sašo Jezernik, Gery Colombo, Thierry Keller, Hansruedi Frueh, and Manfred Morari. Robotic orthosis lokomat: A rehabilitation and research tool. *Neuromodulation: Technology at the neural interface*, 6(2):108–115, 2003.

[3] Alberto Esquenazi, Mukul Talaty, Andrew Packel, and Michael Saulino. The rewalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury. *American journal of physical medicine & rehabilitation*, 91(11):911–921, 2012.

[4] Jonathan C Mcleod, Susie JM Ward, and Audrey L Hicks. Evaluation of the keeogo^{TM}dermoskeleton. *Disability and Rehabilitation: Assistive Technology*, pages 1–10, 2017.

[5] Kai Schmidt and Robert Riener. Maxx: mobility assisting textile exoskeleton that exploits neural control synergies. In *Converging Clinical and Engineering Research on Neurorehabilitation II*, pages 539–543. Springer, 2017.

[6] Alexander Duschau-Wicke, J v Zitzewitz, L Lünenburger, and Robert Riener. Patient-driven cooperative gait training with the rehabilitation robot lokomat. In *4th European Conference of the International Federation for Medical and Biological Engineering*, pages 1616–1619. Springer, 2009.

[7] Sai K Banala, Alexander Kulpe, and Sunil K Agrawal. A powered leg orthosis for gait rehabilitation of motor-impaired patients. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 4140–4145. Citeseer, 2007.

[8] Bram Koopman, Edwin HF van Asseldonk, and Herman van der Kooij. Speed-dependent reference joint trajectory generation for robotic gait support. *Journal of biomechanics*, 47(6):1447–1458, 2014.

[9] S Focke Martinez, Olena Kuzmicheva, and Axel Gräser. Prediction of characteristic points of hip and knee joint trajectories during overground walking using imus and artificial neural networks. In *Medical Measurements and Applications (MeMeA), 2016 IEEE International Symposium on*, pages 1–6. IEEE, 2016.

[10] Youngmok Yun, Hyun-Chul Kim, Sung Yul Shin, Junwon Lee, Ashish D Deshpande, and Changhwan Kim. Statistical method for prediction of gait kinematics with gaussian process regression. *Journal of Biomechanics*, 47(1):186–192, 2014.

[11] Nicholas B Melo, Carlos ET Dorea, Pablo J Alsina, and Márcio V Araujo. Joint trajectory generator for powered orthosis based on gait modelling using pca and fft. *Robotica*, 36(3):395–407, 2018.

[12] Saso Jezernik, Gery Colombo, and Manfred Morari. Automatic gait-pattern adaptation algorithms for rehabilitation with a 4-dof robotic orthosis. *IEEE Transactions on Robotics and Automation*, 20(3):574–582, 2004.

[13] Fares J Abu-Dakk, Angel Valera, Juan A Escalera, Marina Vallés, Vicente Mata, and Mohamed Abderrahim. Trajectory adaptation and learning for ankle rehabilitation using a 3-prs parallel robot. In *International Conference on Intelligent Robotics and Applications*, pages 483–494. Springer, 2015.

[14] Letian Wang, Edwin HF van Asseldonk, and Herman van der Kooij. Model predictive control-based gait pattern generation for wearable exoskeletons. In *Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on*, pages 1–6. IEEE, 2011.

[15] Auke Jan Ijspeert, Jun Nakanishi, and Stefan Schaal. Movement imitation with nonlinear dynamical systems in humanoid robots. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 2, pages 1398–1403. IEEE, 2002.

[16] Heike Vallery, Edwin HF Van Asseldonk, Martin Buss, and Herman Van Der Kooij. Reference trajectory generation for rehabilitation robots: complementary limb motion estimation. *IEEE transactions on neural systems and rehabilitation engineering*, 17(1):23–30, 2009.

[17] Pu Duan, Shilei Li, Zhuoping Duan, and Yawen Chen. Bio-inspired real-time prediction of human locomotion for exoskeletal robot control. *Applied Sciences*, 7(11):1130, 2017.

[18] Tingfang Yan, Andrea Parri, Virginia Ruiz Garate, Marco Cempini, Renaud Ronsse, and Nicola Vitiello. An oscillator-based smooth real-time estimate of gait phase for wearable robotics. *Autonomous Robots*, 41(3):759–774, 2017.

[19] Eunsuk Chong and Frank Chongwoo Park. Movement prediction for a lower limb exoskeleton using a conditional restricted boltzmann machine. *Robotica*, 35(11):2177–2200, 2017.

[20] Cornelius Glackin, Christoph Salge, Martin Greaves, Daniel Polani, Siniša Slavnić, Danijela Ristić-Durrant, Adrian Leu, and Zlatko Matjačić. Gait trajectory prediction using gaussian process ensembles. In *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*, pages 628–633. IEEE, 2014.

[21] Marko Ackermann and Antonie J. van den Bogert. Optimality principles for model-based prediction of human gait. *Journal of Biomechanics*, 43(6):1055 – 1060, 2010.

[22] Lei Ren, Richard K Jones, and David Howard. Predictive modelling of human walking over a complete gait cycle. *Journal of biomechanics*, 40(7):1567–1574, 2007.

[23] Yujiang Xiang, HJ Chung, A Mathai, S Rahmatalla, J Kim, T Marler, S Beck, J Yang, JS Arora, K Abdel-Malek, et al. Optimization-based dynamic human walking prediction. Technical report, SAE Technical Paper, 2007.

[24] Jan Rueterbories, Erika G Spaich, Birgit Larsen, and Ole K Andersen. Methods for gait event detection and analysis in ambulatory systems. *Medical engineering & physics*, 32(6):545–552, 2010.

[25] Juri Taborri, Eduardo Palermo, Stefano Rossi, and Paolo Cappa. Gait partitioning methods: A systematic review. *Sensors*, 16(1):66, 2016.

[26] Huong Vu, Felipe Gomez, Pierre Cherelle, Dirk Lefeber, Ann Nowé, and Bram Vanderborght. Ed-fnn: A new deep learning algorithm to detect percentage of the gait cycle for powered prostheses. *Sensors*, 18(7):2389, 2018.

[27] Jonas Vantilt, Kevin Tanghe, Maarten Afschrift, Amber KBD Bruijnes, Karen Junius, Joost Geeroms, Erwin Aertbeliën, Friedl De Groote, Dirk Lefeber, Ilse Jonkers, et al. Model-based control for exoskeletons with series elastic actuators evaluated on sit-to-stand movements. *Journal of neuroengineering and rehabilitation*, 16(1):65, 2019.

[28] C. Vergara, Joris De Schutter, and Erwin Aertbeliën. Combining imitation learning with constraint-based task specification and control. *IEEE Robotics and Automation Letters*, 4(2):1892–1899, 2019.

[29] Erwin Aertbeliën and Joris De Schutter. Learning a predictive model of human gait for the control of a lower-limb exoskeleton. In *5th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics*. IEEE, 2014.

[30] Marta Moltedo, Tomislav Bacek, Karen Junius, Bram Vanderborght, and Dirk Lefeber. Mechanical design of a lightweight compliant and adaptable active ankle foot orthosis. In *Biomedical Robotics and Biomechatronics (BioRob), 2016 6th IEEE International Conference on*, pages 1224–1229. IEEE, 2016.

[31] Marta Moltedo, Tomislav Bacek, Kevin Langlois, Karen Junius, Bram Vanderborght, and Dirk Lefeber. A compliant lightweight and adaptable active ankle foot orthosis for robotic rehabilitation. In *Wearable Robotics: Challenges and Trends*, pages 45–49. Springer, 2017.

[32] Samuel R Hamner, Ajay Seth, and Scott L Delp. Muscle contributions to propulsion and support during running. *Journal of biomechanics*, 43(14):2709–2716, 2010.

[33] OpenSim. *Public available musculoskeletal model*, 2009 (accessed December 3, 2018).

[34] Friedl De Groote, Tinne De Laet, Ilse Jonkers, and Joris De Schutter. Kalman smoothing improves the estimation of joint kinematics and kinetics in marker-based human gait analysis. *Journal of biomechanics*, 41(16):3390–3398, 2008.

[35] Stefan Lambrecht, Anna Harutyunyan, Kevin Tanghe, Maarten Afschrift, Joris De Schutter, and Ilse Jonkers. Real-time gait event detection based on kinematic data coupled to a biomechanical model. *Sensors*, 17(4):671, 2017.

[36] Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.

[37] MATLAB statistics and machine learning toolbox. https://www.mathworks.com/help/stats/ppca.html, 2013–2019.

[38] Christopher V Rao, James B Rawlings, and Jay H Lee. Constrained linear state estimation - a moving horizon approach. *Automatica*, 37(10):1619–1628, 2001.

[39] Peter Kühl, Moritz Diehl, Tom Kraus, Johannes P Schlöder, and Hans Georg Bock. A real-time algorithm for moving horizon state and parameter estimation. *Computers & chemical engineering*, 35(1):71–83, 2011.