Fuzzy forward dynamics of distinct gait phases with a prosthetic foot

Eduard S. Scheiterer · Sigrid Leyendecker

Received: 6 October 2021 / Accepted: 19 March 2022 / Published online: 22 May 2022 © The Author(s) 2022

Abstract
This work considers epistemic uncertainty in the form of fuzzy parameters in a multibody forward dynamics simulation of the human leg with a prosthetic foot. The thigh and shank are modelled as rigid bodies while the prosthetic foot, modelled after a carbon spring prosthesis, is represented by a predeformed geometrically exact beam model. A variational integrator is used to solve the equations of motion in the forward dynamics simulation and the Graph Follower algorithm is used to include epistemic uncertainty. Two cases are considered. Large movements are examined using a pendulum swing, similar to the swing phase during human gait. To validate the deformation of the prosthesis, a second case is examined, where the prosthesis is fixed in space and deforms under the weight of the body segments. Both cases consider a fuzzy Young’s modulus and determine the envelopes of the resulting uncertain target output. The Graph Follower algorithm was modified to substantially increase computational efficiency, enabling the propagation of fuzzy uncertainty through the complex multibody model with rigid and flexible bodies.

Keywords Epistemic uncertainty · Fuzzy forward dynamics · Human gait · Predeforemed geometrically exact beam · Carbon spring prosthesis · Graph Follower algorithm

1 Introduction
The ability to walk upright is a key aspect of human locomotion. Losing this ability due to injury or illness has a significant impact on the life of the unfortunate human. For instance, Waters and Mulroy [1] examines the effect of various illnesses and gait impairments on the effort it takes to walk at different speeds and [2] summarises the causes for amputation and examines different lower limb prosthesis designs and examines future trends. Studying the effects of impairments on gait and improving the design of prosthetic feet is an ever current field of research [3], with the oldest prosthetic limbs dated around 2700 BC [4]. The main goal of lower limb prosthetic design is to mimic the biological counterpart in functionality and behaviour, while at the same time, respecting weight, size and visual appearance constraints. Apart from providing the same functionality, the prosthetic limb should also reduce the psychological effect of the prosthesis on the patient [2]. This makes prosthesis design a complex and interdisciplinary endeavour. As with most engineering design processes, accurate simulation provides a powerful tool in this design process.

To accurately simulate a given system, it is necessary for the model to represent the key aspects of the system. This also requires precise knowledge of the parameters used to describe the model. However, precise determination of the model’s parameters requires effort and may not always be possible. How much effort is put into the model and the precise determination of parameters, of course, depends on the goal of the simulation. This requires for a balance to be found, between accuracy and available computing power. In case of the human leg with a prosthetic foot during gait, the model has to represent the movement of the different body parts while also capturing the deformation energy stored in the prosthetic foot. In this work, thigh and shank are modelled as rigid bodies in the director formulation, see [5], while the prosthetic foot is modelled as a predeformed geometrically exact beam, as seen in [6]. However, most simulations rely on deterministic inputs which cannot always be obtained or require enormous measurement effort to approximate [7–10]. This means that when simulating, uncertainties have to be taken into account. These uncertainties can be subdivided into model and data uncertainty and can further be classified...
by the cause for the uncertainty. If the uncertainty is caused by incompleteness or impreciseness, it is classified as epistemic uncertainty and can be represented with fuzzy approaches. If the uncertainty is caused by variance it is classified as aleatoric uncertainty and can be represented with stochastic approaches [11]. To account for epistemic uncertainty, the Graph Follower algorithm developed in [12] is used in this work. In [12–14] the Graph Follower algorithm is applied to a multi-story frame with three stories. This system, a model commonly used in structural analysis, see for instance [15], has three degrees of freedom. In this work, the human leg is modelled as a multibody system with a geometrically exact beam prosthetic foot which results in a system with many degrees of freedom. This increase in degrees of freedom and the addition of predeformed geometrically exact beam theory into the multibody system result in a highly non-linear and complex system. To compensate for this increase in the complexity and lower the computational cost of propagating fuzzy uncertainty through this system, modifications to the Graph Follower algorithm are proposed to greatly increase its efficiency.

The main contribution of this work are the efficiency increasing modifications to the Graph Follower algorithm, which enable its use to propagate epistemic uncertainty through much more complex systems. This is then demonstrated for a highly non-linear forward dynamics simulation of a complex multibody model with rigid and flexible bodies. This leads to the following structure of this work. The multibody model with rigid and flexible bodies of the human leg and a prosthetic foot is detailed before summarising the Graph Follower algorithm. Then, the modifications made to the Graph Follower algorithm to substantially increase its efficiency such that it can be applied to the complex system of the human leg with a geometrically exact beam prosthetic foot model are described, followed by the examination of two configurations of the model and the discussion of the results.

2 Modelling the human leg with a prosthetic foot

This work focuses on the propagation of epistemic parameter uncertainty in a forward dynamics simulation of the human leg with a prosthetic carbon spring foot. To this end, a suitable model of the human leg with a deformable prosthetic foot is developed first. This model has to allow for the leg’s movement and has to represent the deformation energy of the prosthesis as accurately as possible while not incurring a too high computational cost. The leg is modelled as a kinematic chain, i.e. a series of rigid bodies (thigh, shank) and flexible bodies (prosthetic foot) connected by joints using the director formulation from [16].

2.1 The leg model

The leg consists of the thigh, the shank and the prosthetic foot connected by joints. The thigh and shank are assumed to be rigid bodies while the prosthesis can deform. The hip is modelled as a spherical joint between the thigh and a fixed reference frame. The knee joint is reduced to a single degree of freedom, the rotation about one axis, and is thus modelled as a revolute joint allowing for the primary movement occurring during normal gait in the sagittal plane. A rigid body, as well as a cross-section of the beam are represented by a placement $\mathbf{\varphi} \in \mathbb{R}^3$ and an orthonormal director triad $d_1, d_2, d_3 \in \mathbb{R}^3$, see the middle of Fig. 1.

Orthonormality of the directors is enforced via so called internal constraints, representing the rigidity of the bodies or of the cross-sections respectively. The interconnection between the rigid and flexible bodies and with the fixed reference frame gives rise to external constraints representing the joints. Joint constraint formulations are explained in detail in [5,16]. In this work, parameters based on [18] are used for mass, inertia and size of the thigh and shank and are summarised in Table 1. The prosthetic foot is modelled after the Össur Vari-Flex® carbon spring prosthesis [19]. The prosthetic foot is rigidly attached to the shank, mimicking a passive prosthetic foot. This means that there is no actuator inputting energy into the prosthetic foot during gait, opposed to for instance the “Empower Prothesenfuss” from [20], which actively controls the prosthesis during gait and can reduce metabolic cost for the wearer [2,21]. However, such an active prosthesis requires external energy. Here, a passive prosthetic foot is examined, which only stores and releases energy provided by the patient.

2.2 The carbon spring prosthesis model

In this work, the passive prosthetic foot Össur Vari-Flex® from [19], shown in the top of Fig. 1, is modelled in [22,23] with geometrically exact beam theory from [6,24]. This formulation allows for accurate representation of complex strain states in the prosthesis while still keeping the computational demand relatively low compared to full 3D finite element models. This is crucial for the fuzzy forward dynamics simulation, since the model has to accurately represent the prosthesis’ deformation energy while still keeping the computational cost low enough to make the many forward dynamic evaluations required in the uncertainty propagation algorithm feasible. The location of the centerline of the beam is denoted by $\mathbf{\varphi}(s, t) \in \mathbb{R}^3$, where $s$ runs along the beam’s length and $t$ denotes time. The orientation of a given cross-section is represented by an orthonormal director triad $d_1(s, t), d_2(s, t), d_3(s, t) \in \mathbb{R}^3$. Due to the orthonormality, the cross-sections of the beam are modelled as rigid sections, omitting warping and distortion effects in the
The prosthesis deformation energy is modelled with a St. Venant-Kirchhoff-type stored energy function [25,26] which describes an ideally elastic material behaviour.

\[ W_{int}(\Gamma, K) = \frac{1}{2} \int_0^L \left( \Gamma^T \cdot D^\Gamma \cdot \Gamma + K^T \cdot D^K \cdot K \right) \]  

(1)

The material parameters and geometry of the beam are included in the two matrices \( D^\Gamma = \text{diag}(G A, G A, E A) \) and \( D^K = \text{diag}(E I_1, E I_2, G J) \) consisting of the Young’s modulus \( E \), the shear modulus \( G \) and geometry specific parameters e.g. the cross-section area \( A \), the area moments of inertia \( I_1, I_2 \) and the polar moment of inertia \( J \). The values derived via reverse engineering in [23] are summarised in Table 2.

The strain measures \( \Gamma(q) \) and \( K(q) \) quantify shear, elongation, flexion and torsion with

\[
\Gamma_i = d_i^T \cdot \frac{d}{ds} \varphi \left|_{i_0} \right. \\
K_i = \frac{1}{2} \epsilon_{ijk} \left[ d_k^T \cdot \frac{d}{ds} d_j \right] \left|_{i_0} \right.,
\]

(2)

where \( i, j, k = 1, 2, 3 \). Figure 1 shows the prosthesis in the top, a geometrically exact beam with the centerline and directors in the middle, while the resulting model of the prosthesis from [22,23] in an undeformed and a deformed state, using the predeformed geometrically exact beam formulation for the prosthesis geometry, is shown in the bottom.

To complete the model, viscous damping is included in the beam model. Following [25–27], the effective material constitutive equations are extended by a viscous term, proportional to the material strain rates \( \dot{\Gamma}_i, \dot{K}_i \), with viscosity parameters for extension \( \eta_E \) and shear \( \eta_G \). Similar to the energy function above, the Kelvin-Voigt viscous dissipative potential \( V \) is defined as

\[ V = \frac{1}{2} \int_0^L \left( \dot{\Gamma}^T \mathbf{D}^{\eta_1} \dot{\Gamma} + \dot{K}^T \mathbf{D}^{\eta_2} \dot{K} \right) \]

(4)

where \( \mathbf{D}^{\eta_1} = \text{diag}(\eta_E A_1, \eta_G A_2, \eta_E A) \) and \( \mathbf{D}^{\eta_2} = \text{diag}(\eta_E I_1, \eta_E I_2, \eta_G J) \) combine material and geometric parameters, as for instance in [26]. Since no information on the viscous damping parameters \( \eta_E \) and \( \eta_G \) for the prosthesis are currently available, this work uses \( \eta_E = \eta_G = 1.0 \), to counteract high frequency vibrations inside the beam. This continuous formulation of the beam can be discretised using one-dimensional finite elements, see for instance [28,29], and solved numerically. In this work, the prosthesis is discretised with 20 nodes resulting in 19 one dimensional finite elements, striking a balance between accuracy and computational cost. The choice of 20 spatial nodes is based on a convergence study building on the work detailed in [30] for the conservative case. The viscous damping introduced to the model is included in the convergence study here and the results are shown in Fig. 2. The prosthesis’ highest total deformation energy \( W_{int} \) during a 1 second simulation in a gravitational field is used as a comparison value when varying the number of nodes \( n \) used to discretise the prosthesis. Figure 2 shows the convergence of the internal deformation energy error \( \epsilon_n \) relative to a reference solution (computed on a very fine grid of \( n_{\text{max}} = 250 \) nodes), when increasing the number of nodes used to discretise the prosthesis. The error is calculated for each considered number of nodes \( n \).

\[
\epsilon_n = \frac{||W_{int,n} - W_{int,n_{\text{max}}}||}{||W_{int,n_{\text{max}}}||}
\]

(5)

Increasing the number of nodes leads to an increase in the forward dynamics computation time. Due to the impact the computation time of the forward dynamics has on the fuzzy forward dynamics simulation, 20 nodes are chosen as a balance between accuracy and computational cost in this work.

| Table 1 Parameters for the thigh and shank used in this works simulations, from [18] |
|---------------------------------|----------------|---------|
| Length                         | 0.4582         | [m]     |
| Mass                           | 6.5233         | [kg]    |
| Inertia J1                     | 0.1137         | [kg/m²] |
| Inertia J2                     | 0.1158         | [kg/m²] |
| Inertia J3                     | 0.0225         | [kg/m²] |
| Centre of mass distance to proximal centroid | 0.1779 | [m] |

| Table 2 Isotropic material parameters for the prosthesis derived in part with reverse engineering in [23] along with the assumed viscous damping parameter |
|---------------------------------|----------------|---------|
| Young’s modulus \( E \)         | 59,958,643     | [N/m²]  |
| Poisson’s ratio \( \nu \)       | 0.265          |         |
| Shear modulus \( \tau \)       | 23.693,643     | [N/m²]  |
| Density \( \rho \)              | 1.71,693,643   | [kg/m³] |
| Viscous damping \( \eta \)      | 1.0            |         |
Fig. 1 The Össur Var-Flex® (top) which is modelled with the geometrically exact beam theory (middle, based on [24]) resulting in the high fidelity predeformed prosthesis model in Matlab (bottom) in the reference configuration (transparent) and in a deformed configuration (coloured by stress resultants). The image showing the Össur Var-Flex prosthesis (top) needs to be excluded from the Creative Commons License (Source: currently [19]). (Color figure online)

This examination is based on the forward dynamics simulation, detailed in the next section. Together with the two rigid bodies, the choice of 20 nodes for the prosthesis’ discretisation leads to a total of 264 degrees of freedom for the model, constrained by 132 internal constraints and 14 external constraints (three for the hip, five for the knee and six for the ankle joint).

2.3 Forward dynamics simulation

To examine the behaviour of the leg with the prosthesis before including parametric uncertainty, the model is placed in a gravitational field orientated in the negative $z$-direction and held in place at the hip with a spherical joint. The resulting motion of the simulation is a swinging motion of the human leg model with prosthetic foot, similar to the swing phase during the human gait cycle. A variational integrator is used to approximate the dynamics, derived from the discrete action principle for the discretised augmented Lagrangian, see [5]. All derivatives and gradients in this work are calculated automatically with Casadi [31]. For the simulation shown in Fig. 3, a timestep of 0.001 s is used. Furthermore, the Young’s modulus of the prosthesis is reduced and its density is increased, compared to carbon fibre laminate, in order to emphasise its deformation due to inertia. However, the simulation also runs stably with the original material parameters. As can be seen in Fig. 4 the energy is exchanged between the potential, kinetic and internal deformation energy. Due to the variational integrator, which is structure preserving, the total energy of the system is not artificially reduced or increased by numerical integration. The viscous damping is present, however, its influence is so small that it is not visible in the energy evolution plot.

3 Fuzzy forward dynamics

We now consider the propagation of epistemic uncertainty in the form of fuzzy model parameters through a forward
μ(\text{membership function})

A fuzzy number \( \tilde{F} \) can be defined as a function of an interval that defines its possible values and its fuzzy numbers is necessary. A fuzzy number \( \tilde{F} \) can be described, a basic understanding of the Graph Follower algorithm performed in this work can be described. A modified version of the Graph Follower algorithm developed in [12] is used to compute the uncertainty propagation through the forward dynamics simulation. Before the modifications to the Graph Follower algorithm performed in this work can be described, a basic understanding of the Graph Follower algorithm and epistemic uncertainty in the form of fuzzy numbers is necessary. A fuzzy number \( \tilde{p} \) is characterised by an interval that defines its possible values and its membership function \( \mu(\tilde{p}) \), see for instance [32].

\[ \mu_p(p_{\text{det}}) = 1 \]

resulting in a triangular shape for the membership function.

Propagating this uncertain parameter through a forward dynamics simulation has the goal to calculate both the resulting interval of a target output quantity \( f_d(\tilde{p}) \) as well as the corresponding membership function \( \mu_f \) at any time, resulting in the fuzzy target output \( \tilde{f}_d(\tilde{p}) \). This results in an optimisation problem to find the extremes of the target output given the uncertain input interval. The target output \( f_d \) represents a scalar quantity of interest that is calculated from the model’s trajectory \( q(\tilde{p}, t) \) and the uncertain parameters \( \tilde{p} \). The uncertain parameters \( \tilde{p} \) either directly affect the target output or indirectly affect it, by only influencing the forward dynamics \( q(\tilde{p}, t) \).

\[ f_d = F(q(\tilde{p}, t), \tilde{p}, t) \]

(6)

To compute this optimisation on a computer, a discretisation of the problem is necessary. Using \( \alpha \)-level cuts, it is possible to discretise the membership function and intervals, shown in the middle of Fig. 5 for the the \( \alpha \)-level cuts \( \alpha = 0.0 \) and \( \alpha = 0.5 \). For \( \alpha = 1.0 \) the interval of the fuzzy parameters reduce to a single parameter, \( p_{\text{det}} \). For more details on this discretisation see for instance [7,8]. The Graph Follower algorithm computes the target output envelopes using \( \alpha \)-level optimisation. This means the intervals of the target output function are calculated with an optimisation based on the input’s fuzzy parameter interval, while the membership function is inferred from the chosen \( \alpha \)-level of the uncertain input parameter, see [8]. This can be done for a singular uncertain parameter or multiple, however, depending on the interaction of these parameters, further unintentional uncertainty may be introduced [8]. The method of propagating fuzzy uncertainty with \( \alpha \)-level optimisation is based on the extension principle for fuzzy numbers [32] and Nguyen’s note on the extension principle [33] and is detailed in e.g. [7,8] and only summarised briefly here. The goal of propagating a fuzzy uncertain parameter through a model is to calculate the fuzzy target output \( \tilde{f}_d \), which means both the numerical values \( f_d \) as well as the associated membership function values \( \mu_f \). These can be calculated with \( \alpha \)-level optimisation which discretises the input fuzzy numbers with \( \alpha \)-level cuts and then performs an optimisation to find the extreme values of the target output \( f_{d,\alpha k} \) for the given \( \alpha \)-level cut \( \alpha_k \). When the two extreme values of the target output at a given timestep are known for a given \( \alpha \)-level cut, two points of the target output membership function \( \mu_f \) are known since the values are inherited from the membership function value of the \( \alpha \)-level cut due to Nguyen’s principle. The target output fuzzy number \( \tilde{f}_d \) can be computed by solving the following optimisation problem for each timestep and multiple \( \alpha \)-level

\[ \mu_p(p_{\text{min}}) = \mu_p(p_{\text{max}}) = 0 \]
cuts as described for instance in [12–14].

\[ f_{d, \alpha_k}(\tilde{\mathbf{p}}^s, \alpha_k) = -s \min_{\mathbf{p} \in \mathcal{P}_{\alpha_k}} (-s f_d(\mathbf{p})) \quad \text{with} \quad s = \mp 1 \quad (7) \]

Solving this optimisation problem includes the evaluation of the deterministic mapping of the target function \( f_d \) and thus for this work the forward dynamics simulation. The Graph Follower algorithm combines the \( \alpha \)-level optimisation with an approximation of the forward dynamics simulation via Taylor expansion and post-processing steps to efficiently calculate the target output envelope and the target output’s membership function over time, resulting in an approximation of the fuzzy target output. The process of forward propagation of epistemic uncertainty to calculate a target output envelope is visualised in Fig. 5.

The Graph Follower algorithm from [12] employs two main methods to make \( \alpha \)-level optimisation feasible with forward dynamics. The first is the linearisation of the forward dynamics with a Taylor expansion with respect to the last known optimal parameter, allowing for quick computations of the target output at the cost of accuracy. The second method, is the combination of storage and post-processing. When a new parameter is found by the optimisation, the forward dynamics are calculated with it and the target output is computed for the entire simulation time. Both the parameter as well as its associated target output evolution are stored in a library within one \( \alpha \)-level. This library of previously computed parameters and target output trajectories is used to choose a parameter for the linearisation of the forward dynamics based on the currently known extreme target output trajectory. The parameter \( \mathbf{p} \) that is associated with the currently known extreme target output trajectory is used for the linearisation to approximate the forward dynamics and allow an efficient optimisation. Furthermore, at the end of an \( \alpha \)-level optimisation for a given \( \alpha \)-level, all target output trajectories are compared and only the extreme values are transferred to the target output envelope. This is repeated with several \( \alpha \)-levels, to define the membership function value for the envelopes.

3.2 Fuzzy parameters of the prosthesis model

The prosthesis’ Young’s modulus is assumed to be subject to uncertainty. Alternative uncertain parameters could be the body segment length, joint locations, other material parameters for the prosthesis or any combination of these parameters. However, since this work’s goal is to demonstrate the propagation of epistemic uncertainty through a complex nonlinear model and to allow for an easy interpretation of the results, only the Young’s modulus is assumed to be uncertain here before examining multiple uncertain parameters. The prosthesis’ Young’s modulus deterministic value \( E_{det} \) is subjected to epistemic uncertainty by modelling it as a triangular fuzzy number as \([0.9, 1.0, 1.1] \cdot E_{det} \), shown exemplary in the middle of Fig. 5. The fuzzy Young’s modulus is discretised with three \( \alpha \)-levels: \([1.0, 0.5, 0.0] \), where \( \alpha = 1.0 \) is expected to yield the same results as the deterministic solution for triangular input fuzzy numbers, while \( \alpha = 0.0 \) allows for the largest variance in the parameter. It should be noted, that the parameter values in the interval have to lead to solutions for the forward dynamics simulation before the Graph Follower algorithm can be applied.

3.3 Speeding up the Graph Follower algorithm

This section details the modifications made to the Graph Follower algorithm in order to achieve the shown results. These modifications are necessary to keep it computationally feasible for the complexity of the analysed system.

3.3.1 Reduction of evaluation points

To accomplish this, the timestep of the forward dynamics and the Graph Follower algorithm are separated. To compute stably, the forward dynamics of the leg with a geometrically exact prosthetic foot requires a timestep of 0.001 s. Using the same timestep for the Graph Follower algorithm for a simulation time of 2.0 s would result in 2000 \( \alpha \)-level optimisations for each discrete \( \alpha \)-level envelope, meaning once for the upper and once for the lower boundary, for every examined \( \alpha \)-level. Additionally, for every new optimal parameter found by the Graph Follower algorithm, a new forward dynamics simulation is necessary to calculate the associated target output. In the worst case, this would lead to \( n_{\alpha\text{-levels}} \cdot n_{\text{timesteps}} \cdot 2 \) forward dynamics simulations, which is computationally expensive.

To reduce this, the \( \alpha \)-level optimisation is only performed every 100 steps of the forward dynamics simulation. This leads to greatly reduced computational cost. The resulting envelopes are still calculated with the forward dynamics timestep of 0.001 s, resulting in smooth trajectories. The number of required optimisations is greatly reduced at the cost of possibly not finding all extreme trajectories. Figure 6 shows the resulting envelopes for two different optimisation timesteps to compare the effects of this reduced optimisation timestep. Table 3 lists the maximum difference between the envelopes. Both simulations were performed using the parameters for the swing configuration, described in detail in the next section. The difference between the two compared optimisation timesteps is negligible for this case.

3.3.2 Tolerance distinguishing optimal parameters

The Graph Follower algorithm maximises or minimises the target output function within the parameter interval. If an
Visualisation of the uncertainty propagation process using the Graph Follower algorithm. The target output quantity $f_d(q(\tilde{p}, t), \tilde{p}, t)$ calculated from the discrete parameter interval values $p_{\text{min}}, p_{\text{det}}, p_{\text{max}}$ leads to distinct evolutions of the target output (left). Including epistemic uncertainty in the form of a convex fuzzy number (middle) which is discretised into $\alpha$-level cuts, the output envelopes of the target output can be calculated with the Graph Follower algorithm (right) resulting in the extreme values for the target output, as well as the corresponding membership function $\mu$ at every time node. (Color figure online)

The resulting envelopes for two different $\alpha$-level optimisation step sizes in the Graph Follower algorithm. The top figure shows the results for an optimisation every 100 forward dynamics simulation, while the middle shows the results for an optimisation every 10 timesteps. The bottom shows the difference between the envelopes for the three examined $\alpha$-levels. (Color figure online)

| $\alpha$-level | $\Delta H_L$ | $\Delta H_U$ |
|----------------|-------------|-------------|
| 1.0            | 0.0         | 0.0         |
| 0.5            | 0.0110e−11  | 0.7254e−11  |
| 0.0            | 0.0916e−8   | 0.3369e−8   |

The maximum difference between envelopes calculated with reduced optimisation timestep and the same timestep as the forward dynamics simulation, for upper ($H_U$) and lower ($H_L$) envelope.

Optimal parameter is found, it is stored in a library and used for the approximation of the forward dynamics in the optimisation. The last parameter associated with the extreme target output for the current examined timestep is used to linearise the forward dynamics during the optimisation. This parameter library is also used to check whether the new optimal parameter is indeed new, by comparing it to the parameter used for linearisation. A new forward dynamics simulation is only performed if the difference of the optimised parameter to known parameters exceeds a percentual tolerance. This tolerance can be estimated by a sensitivity analysis of the forward dynamics with respect to the uncertain input parameter. To further reduce the computational cost, the forward dynamics simulation is recalculated and the new target output is calculated from the resulting trajectory only if the found optimal parameter is sufficiently different from all previously calculated parameters. Otherwise, the stored values are used. Ensuring the novelty of a parameter within an $\alpha$-level greatly reduces the computational cost of the Graph Follower algorithm.

Of course, this tolerance directly affects the accuracy of the target output envelopes. This means it has to be dimensioned to fit to the examined problem. Also, multiple uncertain parameters make it more difficult to apply, since multiple parameters do not affect the target output without interdependencies. In this work, new forward dynamics calculations are performed if the relative difference $p_{\text{diff},i}$ between the current fuzzy parameter and the closest stored parameter exceeds $p_{\text{tol}}$ of the deterministic value. In the case of a simulation with multiple fuzzy parameters, a new simulation is performed if any individual parameter difference exceeds $p_{\text{tol}}$ of the respective deterministic parameter.

$$p_{\text{diff},i} = \frac{|\tilde{p}_i - p_{\text{Bib},i}|}{p_{\text{det},i}}$$

The tolerance value $p_{\text{tol}} = 0.01\%$ was determined experimentally. For more precision in the choice of this tolerance,
a sensitivity analysis of the target output with respect to the fuzzy parameter can be performed.

### 3.4 Limitations of the Graph Follower algorithm

Before examining the scenarios simulated in this work, the limitations of the Graph Follower algorithm are briefly summarised. The goal of propagating fuzzy input parameters through a model, is to calculate the target output fuzzy number. This includes the interval of the target output \( f_d \) and the associated membership values \( \mu_f \). As detailed above, the Graph Follower algorithm uses \( \alpha \)-level optimisation and a linearisation of the forward dynamics with respect to the fuzzy input parameter to efficiently calculate the target output envelopes for multiple \( \alpha \)-levels, where the membership function is derived from the current \( \alpha \)-level, resulting in an approximation of the target output fuzzy number. The \( \alpha \)-level optimisation with forward dynamics imposes limitations on the accuracy of the resulting envelopes. To work precisely, the \( \alpha \)-level optimisation requires a global optimiser and precise calculation of the target output, see [7]. However, global optimisation is computationally expensive and the accurate calculations of the forward dynamics required for the exact target output calculation are also computationally expensive with such a complex model. Thus, to be feasible, the Graph Follower algorithm only approximates global optimisation, by using Matlab’s `MultiStart` function with ten starting points. This significantly increases the result quality with an acceptable increase in computation time, compared to using `fmincon` with just one starting point. The linearisation of the forward dynamics with respect to the fuzzy parameter introduces more inaccuracy. Both of these limitations can lead to an overlap of \( \alpha \)-levels. The modifications necessary for the Graph Follower algorithm to work with the complex model of the human leg with a deformable prosthesis, also require a trade-off between accuracy and computational cost. The separation of the timestep of the forward dynamics simulation and the \( \alpha \)-level optimisation greatly reduces the amount of necessary optimisations but also reduces the chances to find extremising parameters. Also, the tolerance when comparing new parameters to stored parameters can reduce accuracy if it is chosen to large, however, it greatly reduces the computational cost of the modified Graph Follower algorithm. Nonetheless, the modified Graph Follower algorithm, as is used here, produces an approximation of the target output fuzzy number for the complex multibody system of the human leg with a predeformed geometrically exact beam prosthetic foot. Furthermore, these limitations can be reduced or even become negligible by increasing the available computing power. Also, if the precise membership function value of the target output is not required, the trajectories of the target output of the individual \( \alpha \)-levels can be combined to get an accurate global envelope of the target output, providing information on the extreme values of the target output based on the fuzzy input parameter. Thus, even with limitations, the modified Graph Follower algorithm provides valuable information about how epistemic model parameters affect a desired target output for a highly complex multibody system with flexible and rigid bodies, allowing for a quicker examination of the uncertainties effects than simulating many different sets of parameters.

### 3.5 Simulation scenarios

In the two exemplary scenarios analysed in this work, the leg moves under its weight due to being placed in a gravitational field oriented in the negative \( z \)-direction.

The first examined scenario is a pendulum swing of the multibody system, where the prosthesis deforms due to inertia, as was already used to check the models implementation with forward dynamics, shown as a timelapse in Fig. 3. The pendulum movement is similar to the swing phase occurring during human gait (for more information on the biomechanics of the human leg and the human gait cycle see for instance [17]). The parameters used for the simulation are summarised in Table 4.

The second scenario simulates the leg in a squatting position, that is the thigh is horizontal with the shank vertical. The prosthetic foot is held in place at two nodes. This is done by constraining two nodes to remain in their initial position. However, rotation of the cross-sections at these nodes is permitted. The prosthesis deforms under the leg’s weight periodically. Figure 7 shows the initial configuration of the leg alongside a deformed configuration of the resulting movement. The deformation is due to the mass of the leg in a gravitational field. To emulate the process of realistic load on the prosthesis, the mass of the shank is increased to represent the influence of body weight, while the prosthesis uses the parameters derived with reverse engineering. The parameters used for the simulation are summarised in Table 5.

| Young’s modulus \( E \) | 59.95e4 | [N/m²] |
|-------------------------|---------|--------|
| Poisson’s ration \( \nu \) | 0.265   |        |
| Shear modulus \( G \)    | \( \frac{2E}{3(1+\nu)} \) = 23.69e4 | [N/m²] |
| Density \( \rho \)       | 8.55e3  | [kg/m³] |
| Viscous damping \( \eta \) | 1.0     |        |
| \( \Delta t \)           | 0.001   | s      |
| \( T \)                  | 2       | s      |
| Hip extension \( \frac{\pi}{3} \) |        | rad    |
| Knee flexion \( \frac{\pi}{3} \) |        | rad    |

**Table 4** Prosthesis parameters and initial joint angles used for the fuzzy forward dynamics simulation of the swinging motion
Table 5 Prosthesis parameters and initial joint angles used for the fuzzy forward dynamics simulation of the prosthesis under load

| Parameter                      | Value          |
|--------------------------------|----------------|
| Young’s modulus $E$            | $59.958e7$     |
| Poisson’s ration $\nu$        | 0.265          |
| Shear modulus $G$             | $23.69e7$      |
| Density $\rho$                | $8.55e3$       |
| Viscous damping $\eta$        | 1.0            |
| $\Delta t$                    | 0.001          |
| $T$                           | 2              |
| Hip flexion                   | $\pi$         |
| Knee flexion                   | $\pi$         |
| Shank weight                   | 107            |

The total deformation energy stored and released during gait in the prosthetic foot is related to the burden of gait for a patient, see for instance [1]. This means it is directly related to the walking comfort of the patient with the prosthetic foot and is therefore an important measure for the prosthesis’ design and its acceptance by the patient. Predicting the prosthesis’ behaviour with respect to the target output is important during the design process. However, predicting the movement of a double pendulum is difficult, given its chaotic nature. Adding a flexible body with uncertain stiffness to the end of the double pendulum further complicates the prediction. Including fuzzy uncertainty and a target output that depends on both the behaviour of the multibody model and the fuzzy uncertainty increases the need for algorithmic computation of the desired target output. Figure 8 shows the calculated trajectories for the minimal ($0.9 E_{det}$), deterministic ($E_{det}$) and maximum ($1.1 E_{det}$) values of the prosthesis’ Young’s modulus. The Graph Follower algorithm is used to calculate the target output envelopes. Analysing the uncertainty propagation allows for an efficient calculation of the correlation between the parametric input uncertainty, in this case the Young’s modulus, and the target output, the stored internal deformation energy of the prosthesis. The results are shown in Fig. 9. The target output envelopes for the different alpha levels are shown alongside the deterministic simulation’s result. The $\alpha = 1$-level envelope matches the deterministic solution, as expected. The total deformation energy of the prosthesis oscillates which is represented well by the variational integrator used in the forward dynamics which avoids artificial energy loss due to numerical integration. As shown in the examination of the forward dynamics of this model, the energy drain of the viscous damping is negligible for the chosen parameters. While the input uncertainty is triangular and symmetric, the output envelopes shown in Fig. 9 do not inherit these properties.

The limitations of the Graph Follower algorithm can be seen at $t = 1.4s$ when the upper envelope from $\alpha = 0.0$-level intersects the envelope from $\alpha = 0.5$-level.

However, when considering the entire evolution of the envelopes, the inaccuracy is very localised to parts of the tar-
Fig. 9  Visualisation of the upper and lower envelope during the swinging motion of the human leg with the prosthesis for the target output function being the total stored energy within the prosthesis. The deterministic solution is shown with a dashed line, while the envelopes for the different $\alpha$-levels are shown as full lines. (Color figure online)

Fig. 10  The minimal and maximum envelopes for total internal deformation energy in the prosthesis. (Color figure online)

to very computationally intensive forward dynamics simulations. The load applied to the prosthesis emulates the load of the body during a squat exercise. The envelopes of the target function are shown in Fig. 11. As expected, the prosthesis acts as a spring under the load of the shank. The leg moves up and down periodically, while the uncertain Young’s modulus affects both the amplitude and the period of the oscillation. Since the amplitude and frequency of the oscillating movement of the leg is directly related to the prosthesis’ deformation, the oscillating movement can also be seen in the resulting envelopes for the chosen target function. Artefacts from the simplifications necessary for the Graph Follower algorithm to run with the high number of degrees of freedom of the model are present, but can be reduced by increasing computational power.

4 Multiple fuzzy parameters

After demonstrating the application of the Graph Follower algorithm on the complex multibody system of the human leg with a geometrically exact beam prosthesis with a single fuzzy uncertain parameter, the next step is to examine the structure with multiple uncertain parameters. For this, two aspects are of interest. How does the method perform with regards to the increased number of optimisation parameters and which parameters can be affected with uncertainty.

4.1 Numerical efficiency of Graph Follower algorithm with multiple uncertain parameters

Here a brief evaluation of the Graph Follower algorithm with respect to its numerical efficiency is discussed. A comprehensive evaluation of the Graph Follower algorithm and comparison to other methods can be found in [12]. There, it is shown that the Graph Follower algorithm is suitable for propagating multiple fuzzy parameters since its computing time scales much less with the amount of uncertain parameters than the benchmark algorithm Qua.Si.III. To compare
the modified Graph Follower algorithm used in this work, the runtime for four simulations are compared, one with a single fuzzy parameter and one with 11 fuzzy parameters, each for 1 and 2 s with 1000 and 2000 timesteps respectively. The resulting runtimes are shown in Table 6. This shows, that an increase in the amount of uncertain parameters does not affect the computation time of the simulation strongly. This also means, that focus should be put on reducing the computational time of the forward dynamics simulation, to reduce the computational cost when using the Graph Follower algorithm to propagate fuzzy uncertainty.

### 4.2 Choice of fuzzy parameters

The main limiting factor when choosing which parameters to model as uncertain, is the requirement, that the parameters are independent of each other and without interaction [7]. Interdependence or interaction between fuzzy numbers can negatively affect the $\alpha$-level optimisation by introducing unwanted uncertainty. Possible uncertain parameters are the leg’s segment length, the segment’s mass and moments of inertia, the segment’s initial orientation and the prosthesis material parameters, namely Young’s modulus, density and Poisson’s ratio. Here, affecting the segment length or initial position with uncertainty would lead to fuzzy configurations of the leg. However, the configuration is not independent, since it has to satisfy the internal constraints of the rigid bodies of the segments and the joint constraints and can therefore not be affected by uncertainty without further investigation. The prosthesis’ Young’s modulus, Poisson’s ratio and density parameters can safely be assumed to be independent and are affected with uncertainty. For the leg, its mass is affected with uncertainty, including the moments of inertia. These values are not independent of each other, however the exact distribution of the mass along the leg is unknown for an individual and difficult to measure in vivo, which allows for an independent treatment in this work. This leads to a simulation with a total of 11 uncertain parameters. All fuzzy parameters are discretised with the same $\alpha$-level cuts and for comparability also have the same parameters and interval of $P_{\text{det}} \pm 0.1 P_{\text{det}}$ as the squatting simulation.

### 4.3 Results

As in the previous simulation the movement is an oscillation. With the increased number of fuzzy parameters, the period and magnitude of the oscillations differ more than in the simulation with only the fuzzy Young’s modulus leading to larger envelopes, shown in Fig. 12. Due to the multiple uncertain parameters, visualisation of the parameter distribution and their correlation to the envelopes is no longer trivial. Figure 13 visualises the parameter distribution for the $\alpha = 0$ level with the individual trajectories calculated from the corresponding parameter set and the resulting upper and lower envelope. The left side shows the relative difference with respect to the upper (green) and lower (red) bounds for the parameters with the deterministic parameters in the middle (blue). As can be seen, all of the parameters vary and lead to individually extreme target outputs for a given timestep, shown on the right of Fig. 13. It can also be seen, that for instance the Poisson’s ratio ($\nu$) leading to the envelope varies a lot less than for instance the mass of the thigh segment ($M_i$), giving an indication of which parameters have a strong effect on the target output. Also of note, the moments of inertia for the $z$-direction ($I_{1,3}$) do not have an effect on the forward dynamics, since the movement has no rotation around the $z$-axis. This means, that these parameters can vary without affecting the target output. This results in most trajectories of the simulation shown in Fig. 13 being based on the deterministic value.

### 5 Conclusions

In this work, the forward propagation of epistemic uncertainty on the parameter level is investigated for two cases of a human leg with a geometrically exact beam prosthetic foot in a fuzzy forward dynamics simulation. The leg is modelled as a multibody system with rigid and flexible bodies connected...
by joints. The prosthesis’ Young’s modulus, subject to fuzzy uncertainty, is propagated forward through the model and multiple target outputs are evaluated. Multiple fuzzy input parameters are also successfully propagated through the forward dynamics model. Remarkably, increasing the number of uncertain parameters does not significantly increase affect the computation cost. The main contribution of this work is the adaptation of the Graph Follower algorithm to a highly non-linear and complex multibody dynamics setting with rigid and flexible bodies. To achieve this, the efficiency of the Graph Follower algorithm is increased substantially. The computational cost of the algorithm is reduced by coarsening the evaluation grid and adding a tolerance to identify new extreme parameters during the optimisation. This slightly reduces the accuracy in the resulting envelopes, but allows for the use of the Graph Follower algorithm to propagate an uncertain Young’s modulus through to the leg model with a prosthetic foot. This provides an approximation of the fuzzy number of the target output and it’s evolution during the simulation time.

The forward propagation of fuzzy uncertainty through a multibody model with rigid and flexible bodies, can be extended in multiple directions. The model can be expanded from a forward dynamics model, mimicking gait phases and loads, to an optimal control simulation of the human gait. Furthermore, the uncertainty aspect can be extended. As briefly mentioned in the introduction, there are multiple ways to model uncertainty, divided into aleatoric and epistemic uncertainty, or the combination of the two, polymorphic uncertainty. Developing the forward propagation of polymorphic uncertainty is planned through either forward dynamics and also optimal control.

Acknowledgements This work is partly supported by the German Research Foundation (DFG) as part of the Priority Programme SPP 1886 ‘Polymorphic uncertainty modelling for the numerical design of structures’ (Grant No. LE 1841/4-2).

Funding Open Access funding enabled and organized by Projekt DEAL.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Waters RL, Mulroy S (1999) The energy expenditure of normal and pathologic gait. Gait Posture 9(3):207–231
2. Asif M et al (2021) Advancements, trends and future prospects of lower limb prosthesis. IEEE Access 9:85956–85977. https://doi.org/10.1109/ACCESS.2021.3086807
3. Eisentraudt M, Leyendecker S (2019) Epistemic uncertainty in optimal control simulation. Mech Syst Signal Process 121:876–889
4. Thurston AJ (2007) Paré and prosthetics: the early history of artificial limbs. ANZ J Surg 77(12):1114–1119
5. Leyendecker S, Marsden JE, Ortiz M (2008) Variational integrators for constrained dynamical systems. ZAMM J Appl Math Mech 88:677–708
6. Antman SS (1995) Nonlinear problems of elasticity. Springer, Berlin
7. Möller B, Beer M (2004) Fuzzy randomness: uncertainty in civil engineering and computational mechanics. Springer, Berlin
8. Möller B, Graf W, Beer M (2000) Fuzzy structural analysis using $\alpha$-level optimization. Comput Mech 26(6):547–565
9. Viertl R (1997) On statistical inference for non-precise data. Environmetrics 8(5):541–568
10. Viertl R (1995) Statistical methods for non-precise data. CRC Press, Boca Raton
11. Graf W, Götz M, Kaliske M (2015) Analysis of dynamical processes under consideration of polymorphic uncertainty. Struct Saf 52:194–201
12. Eisentraut M, Leyendecker S (2019) Fuzzy uncertainty in forward dynamics simulation. Mech Syst Signal Process 126:590–608
13. Pivovarov D et al (2019) Fuzzy dynamics of multibody systems with polymorphic uncertainty in the material microstructure. Comput Mech 64(6):1601–1619
14. Price MA, Beckerle P, Frank C (2019) Design optimization in lower limb prostheses: a review. IEEE Trans Neural Syst Rehabil Eng 27(8):1574–1588
15. Ashari EE (2014) Calculating free and forced vibrations of multi-story shear buildings by modular method. Res J Recent Sci 3:83–90
16. Leyendecker S, Betsch P, Steinmann P (2008) The discrete null space method for the energy-consistent integration of constrained mechanical systems. Flexible multibody dynamics. Springer, Part III
17. Whittle MW (2014) Gait analysis: an introduction. Butterworth-Heinemann, Oxford
18. Chandler RF et al (1975) Investigation of inertial properties of the human body. Air Force Aerospace Medical Research Lab Wright-Patterson AFB OH
19. https://www.ossur.com/de-de/prothetik/fusse/vari-flex. Accessed 21 Mar 2021
20. https://www.ottobock.de/prothesen/beinprothesen/prothesenfuesse/empower/. Accessed 07 July 2021
21. Herr HM, Grabowski AM (2012) Bionic ankle-foot prosthesis normalizes walking gait for persons with leg amputation. Proc R Soc B Biol Sci 279(1728):457–464
22. Scheiterer ES (2019) Simulation of a prosthetic foot modelled by a predeformed geometrically exact beam. Master’s Thesis
23. Söhnlein M (2019) Qualifzierung von Simulationsparametern einer Fussprothese durch numerische und experimentelle Modalanalyse. Master’s Thesis
24. Crisfield MA, Jelenic G (1999) Objectivity of strain measures in the geometrically exact three-dimensional beam theory and its finite element implementation. Proc R Soc Lond Ser A Math Phys Eng Sci 455:1125–1147
25. Lang H, Linn J, Arnold M (2011) Multi-body dynamics simulation of geometrically exact Cosserat rods. Multibody Syst Dyn 25:285–312. [https://doi.org/10.1007/s11044-010-9223-x]
26. Lang H, Leyendecker S, Linn J (2013) Numerical experiments for viscoelastic Cosserat rods with Kelvin-Voigt damping. In: Proceedings of the ECCOMAS thematic conference on multibody dynamics
27. Schlägl T, Leyendecker S (2016) Electrostatic–viscoelastic finite element model of dielectric actuators. Comput Methods Appl Mech Eng 299:421–439
28. Zienkiewicz OC, Morgan K (2006) Finite elements and approximation. Courier Corporation, North Chelmsford
29. Fish J, Belytschko T (2007) A first course in finite elements. Wiley, Hoboken
30. Scheiterer ES, Leyendecker S (2021) Predeformed geometrically exact beam model for a dynamic-response prosthesis. PAMM 20(1):e202000152
31. Andersson JAE, Gillis J, Horn G, Rawlings JB, Diehl M (2019) CasADi—A software framework for nonlinear optimization and optimal control. Springer. [https://doi.org/10.1007/s12532-018-0139-4]
32. Zadeh LA (1996) Fuzzy sets. Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh, pp 394–432
33. Nguyen HT (1978) A note on the extension principle for fuzzy sets. J Math Anal Appl 64:369–380
34. Courant R, Friedrichs K, Levy H (1928) Über die partiellen Differentialgleichungen der mathematischen Physik. Math Ann

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.