INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2015; **00**:1–18
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme

Computational vademecums for real-time simulation of surgical cutting in haptic environments

C. Quesada[1], D. González[1], I. Alfaro[1], E. Cueto[1*], F. Chinesta[2]

[1] *Aragon Institute of Engineering Research. Universidad de Zaragoza. Zaragoza, Spain.*
[2] *GeM, UMR CNRS-Ecole Centrale de Nantes, France*

SUMMARY

A new method for the real-time simulation of surgical cutting in haptic environments is presented. It is based on the intensive use of computational vademecums, i.e., a sort of computational parametric meta-model, which is computed off-line and only evaluated on-line. Therefore the necessary time savings are obtained, allowing for feedback responses on the order of kHz. Such a high-dimensional, parametric solution of the problem is computed by employing Proper Generalized Decomposition for the off-line phase of the method, along with X-FEM techniques for the incorporation of the discontinuities in the displacement field after cutting, in the on-line phase. A thorough description of the proposed method, along with examples of its performance in the simulation of corneal surgery, are provided. Copyright © 2015 John Wiley & Sons, Ltd.

Received ...

KEY WORDS: Real time, surgery, model order reduction, proper generalized decomposition, X-FEM.

1. INTRODUCTION

The interest of risk-free training of surgeons by employing computer simulations is well known, albeit their inherent difficulties are also noteworthy. The same can be said for pre-operative planning or intra-operative decision-taking systems. Difficulties arise from the high complexity of the system under consideration (i.e., the human body), its intrinsic non-linear [23] [18] [45] and coupled multi-physical characteristics [2] [21] and even its high dimensionality [4].

A very high effort of research is being paid by a big community of researchers to overcome all these different difficulties. Underneath all them it is the need for real-time response. All these highly complex problems should be solved by nowadays computers (possibly, without resorting to supercomputer facilities, ideally on deployed, handheld devices, such as smartphones or tablets [2]). In general, the term "real time" means different things in different contexts. For instance, in the development of surgery training platforms, equipped with haptic (tactile) peripherals, real time means the need to provide the peripherals with feedback responses in the order of 1 kHz (the free-hand vibration frequency) [23] [24] [10]. If only visual continuity is required, as in cinemas, only some 25 to 30 frames per second are required [34]. For surgery planning systems in which long-term predictions surgery outcome are provided, maybe some minutes are enough [26].

*Correspondence to: Elías Cueto. Aragon Institute of Engineering Research. Universidad de Zaragoza. Edificio Betancourt. María de Luna, s/n, E-50018 Zaragoza, Spain. E-mail: ecueto@unizar.es

Contract/grant sponsor: Spanish Ministry of Economy and Competitiveness. Regional Government of Aragon; contract/grant number: This work has been supported by the Spanish Ministry of Economy and Competitiveness through Grant number CICYT DPI2014-51844-C2-1-R and by the Regional Government of Aragon and the European Social Fund, research group T88. Professor Chinesta is also supported by the Institut Universitaire de France.

Different approaches have been pursued in the last decades to partially, at least, overcome this astringent real-time constraint. Some recent surveys trace this history in detail, see [45] [23] [21]. The most relevant and recent advances in the field can be roughly characterized into two main families of methods. The first one is related to the ease of access to "supercomputing" platforms in nowadays very cheap Graphics Processing Units (GPUs). In this way, explicit dynamic finite element codes with lumped mass matrices are well known to run extremely fast on an element-by-element basis, without the need for assembling (nor inverting) the global tangent stiffness matrix [47] and their implementation on a GPU array is straightforward [36] [60] [58].

On a different setting, there has been an increasing interest in many fields (not only computational surgery) towards the exploitation of reduced-order modeling techniques towards the development of models with a minimal number of degrees of freedom, able to be run under severe real-time constraints. Maybe the first work in this framework is due to Barbič and James [6] [5], that found an efficient way to span the kinematics of Kirchhoff-Saint Venant models with a minimal number of degrees of freedom. These works are based upon already classical works in the field of solid mechanics, see [33] [38] [40]. This framework was later generalized to any hyperelastic model (not necessarily linear elastic) by employing Proper Orthogonal Decomposition (POD, also known as PCA [37] [43] [44]) techniques, see [48] [53] [49] [57]. Essentially, POD techniques rely on the off-line computation of *snapshots* whose covariance matrix provides, through an eigenvalue analysis, the relevant *modes* of deformation of the system.

More recently, Proper Generalized Decomposition (PGD) methods [13] [39] [17] [15] further simplify this approach by eliminating the need for the off-line computation of snapshots of the system. In essence, PGD determines on the fly the best basis for the simulation of the system (although results are often not optimal as in POD) and, notably, allow to solve efficiently high-dimensional problems [56] [32] [62][30].

Recent advances in the field employ PGD to construct off-line *computational vademecums* [14] of the systems, essentially an updating of the original vademecums [7] in which our ancient colleagues compiled known solutions to engineering problems of interest. Thus, it is possible to solve off-line reduced versions of very complex and high-dimensional parametric problems so as to exploit them on-line at extremely high feedback rates. This is the approach followed in [51] [52] [28] [46] [3], see Fig. 1, but also, in a different context, in [54] [32] [39].

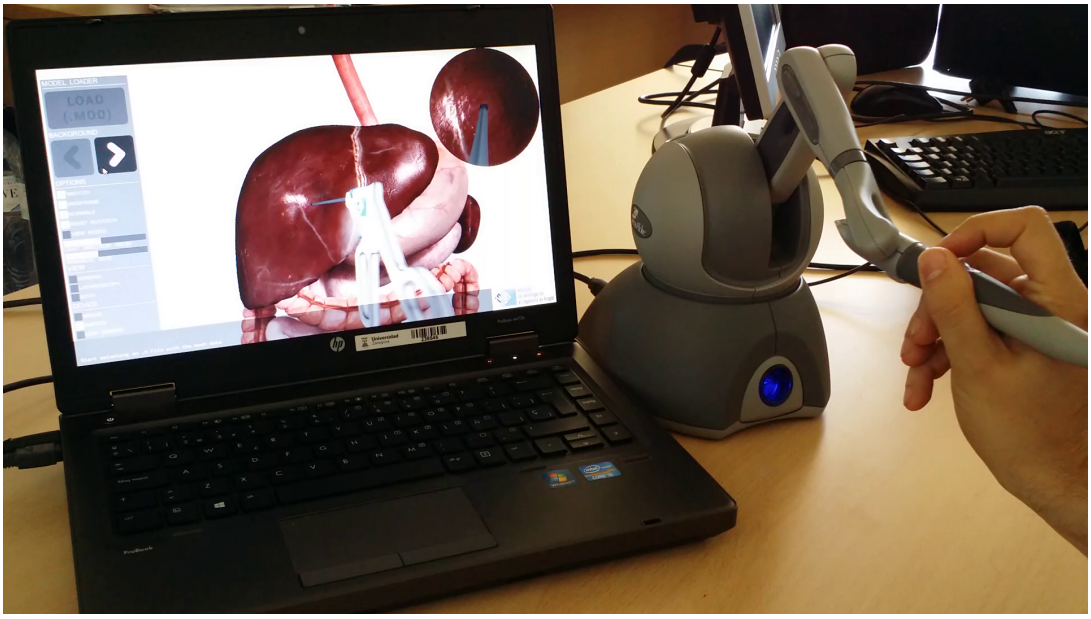

Figure 1. Surgery simulator developed by the authors. On the right, the haptic device can be observed. The simulator runs without any special hardware requirement, on a simple laptop, producing smooth force feedback.

DOI: 10.1002/nme

The problem of simulating surgical cutting is specially challenging [19], since it involves very complex physics [11], but noteworthy, topological changes in the geometry of the organ [19] [8] [20] [31]. Only recently X-FEM-like techniques have been applied to this type of phenomena, see for instance [34] or [50] for a combined employ of X-FEM and POD model reduction techniques. It is also worthy of mention that a meshless approach to this problem has also been developed [35].

In this work we introduce a combination of the concept of computational vademecum and the technique of X-FEM, that greatly simplifies the task of generating and manipulating displacement discontinuities such as those produced by the scalpel cut on the surface of the organ. As will be noticed in Section 4, the simulation is started by the off-line computation of the organ vademecum, i.e., the response of the organ to any possible load produced by the surgical tool on its surface. It is not feasible, however, to pre-compute the result for any possible location and orientation of a cut, or displacement discontinuity. That is why we discuss briefly in Section 2 the usage of PGD modes *à la* POD, i.e., by projection onto the subspace spanned by the PGD separated functions. The possible choices in this regard are discussed.

In the approach here presented, PGD methods have been employed to develop a vademecum, useful when no cut is present, for the simulation of manipulation, palpation, etc. The main novelty in this approach, however, comes into play when a cut is produced (see [11] for a detailed description of the physics prior to the appearance of cutting). At this moment, PGD modes will be used, no more as a vademecum, but as global, Ritz-like shape functions. But these shape functions are parametric, depending on the position of the contact between scalpel and organ. Once this parametric dependence has been particularized for a given position (in our case, detected by the algorithms presented in [28]), an enriched, X-FEM like, model is constructed on the fly so as to take into account the presence of the cut. Results are shown in Section 5 that demonstrate the validity of the technique to be used in haptic environments.

2. ON USING PGD AS IF IT WERE POD: PGD AS A REDUCED BASIS CONSTRUCTOR

A standard application of PGD to the problem at hand will face the difficulty of parameterizing every possible cut (position, length, shape) in the geometry of the organ under consideration. As can be readily noticed, this adds an impressive computational complexity to the problem. However, parametric (continuous) solutions obtained within the PGD framework could be used as reduced basis, which could eventually be enriched by X-FEM techniques so as to be used for performing real time simulations of complex computational mechanics models.

In order to analyze the different possible approaches, and for the sake of simplicity of the exposition, we consider the 1D steady-state heat equation, although the complete hyperelasticity problem will be consider hereafter. Thus, consider

$$\frac{d^2u}{dx^2} - f(x;s) = 0, \quad x \in \Omega = (0,1), \tag{1}$$

with $f(x)$ the source term depending parametrically on the coordinate $s \in \mathcal{S} \subset \Omega$. In the case of a localized source with intensity q, the source term writes $f(x;s) = q\delta(x-s)$. We assume, again for simplicity, homogeneous essential boundary conditions.

The weak form related to problem (1) writes: *find* $u \in \mathcal{H}^1(\Omega)$ *such that*

$$\int_\Omega \frac{du^*}{dx}\frac{du}{dx}\,dx = \int_\Omega u^* f(x;s)\,dx, \tag{2}$$

holds for every $u^* \in \mathcal{H}^1_0(\Omega)$ *verifying homogeneous essential boundary conditions.*

By using standard PGD techniques, widely described in our former works [14], the parametric solution $u(x,s)$ is searched by using a separated representation

$$u(x,s) \approx \sum_{i=1}^{N} X_i(x) \cdot S_i(s). \tag{3}$$

Prepared using nmeauth.cls
DOI: 10.1002/nme

To determine the precise form (i.e., their finite element approximation) of the separated functions X_i and S_i, Eq. (3), we proceed iteratively (see [16]). Suppose that, at iteration $n < N$, the solution $u^n(x, s)$ has the form

$$u^n(x,s) = \sum_{i=1}^{n} X_i(x) \cdot S_i(s). \tag{4}$$

An improvement of the solution at a subsequent iteration $n+1$ consists of looking for the next functional couple $X_{n+1}(x)$ and $S_{n+1}(s)$ involved in the updated solution $u^{n+1}(x, s)$

$$u^{n+1}(x,s) = u^n(x,s) + X_{n+1}(x) \cdot S_{n+1}(s) = \sum_{i=1}^{n} X_i(x) \cdot S_i(s) + X_{n+1}(x) \cdot S_{n+1}(s). \tag{5}$$

For that purpose we consider the *doubly*-weak form

$$\int_{\Omega \times \mathcal{S}} \frac{du^*}{dx} \frac{du}{dx} \, dx \, ds = \int_{\Omega \times \mathcal{S}} u^* f(x;s) \, dx \, ds, \tag{6}$$

with the trial function u^{n+1} given by Eq. (5) and the test function by $u^*(x, y) = X^*(x) \cdot S_{n+1}(s) + X_{n+1}(x) \cdot S^*(s)$, that allows calculating by using an appropriate nonlinear solver the couple of searched functions. Here, X^* and S^* represent, respectively, admissible variations of functions X and S, respectively.

The practical implementation requires writing the source term $f(x; s)$ in a separated form

$$f(x,s) \approx \sum_{j=1}^{M} \mathcal{F}_j(x) \cdot \mathcal{G}_j(s), \tag{7}$$

a task that can be easily accomplished by using the SVD. Note that, if more than one parameter needs to be considered, PGD itself can be employed as a sort of high-order SVD [16].

In what follows we assume that the solution is exactly calculated when using N modes in the separated representation. As discussed in our former works if the computed solution contains too many modes it can be post-compressed by using a SVD. We assume that the calculated solution is optimal in the sense that it cannot be post-compressed anymore and that N is the minimal number of modes for an exact representation of the solution.

Now, as soon as one considers the source located at a certain position s_p, the solution results

$$u(x; s_p) = u^N(x, s_p) = \sum_{i=1}^{N} X_i(x) \cdot S_i(s_p) = \sum_{i=1}^{N} \beta_i^p X_i(x), \tag{8}$$

where the coefficients β_i^p coincide with $S_i(s_p)$.

When considering a different source location p', the solution writes

$$u(x; s_{p'}) = u^N(x, s_{p'}) = \sum_{i} X_i(x) \cdot S_i(s_{p'}) = \sum_{i} \beta_i^{p'} X_i(x), \tag{9}$$

that proves that the space functions are always the same while the coefficients β_i change. Thus, we can conclude that any solution related to any location of the source term can be approximated with the same reduced basis defined by the space functions $X_1(x), \ldots, X_N(x)$. The parametric PGD can be viewed as a constructor of reduced basis in a transparent way for the user.

2.1. On the online projection procedures

There exist, however, two different alternatives to construct the set of reduced basis. Notably, PGD provides the analyst with space and parameter basis functions, so two possibilities arise: (i) to consider a space-only set of functions, by particularizing the parameter value or (ii) to work with the whole space-parameter set of functions. We analyze them in this Section.

DOI: 10.1002/nme

2.1.1. Space reduced basis —srb— Since the space reduced basis can exactly approximate any solution related to any source location, we consider as a particular case the on-line analysis of $s = s_p$,

$$u(x; s_p) = \sum_{i=1}^{N} \beta_i^{p,\text{srb}} \cdot X_i(x), \tag{10}$$

that introduced in the standard weak form, Eq. (2), with

$$u^* = \sum_{i=1}^{N} \beta_i^* \cdot X_i(x), \tag{11}$$

allows calculating the beta coefficients $\beta_i^{p,\text{srb}}$, $i = 1, \ldots, N$, after solving the linear discrete problem of size N. This problem can be viewed as a discretization using the Ritz formulation where functions $X_i(x)$ are optimal in the sense that they can approximate any solution associated with any source location.

If for the ease of exposition we consider that the reduced basis consists of a single function $X(x)$, the coefficient $\beta^{p,\text{srb}}$ involved in the exact solution $u(x, s_p) = \beta^{p,\text{srb}} X(x)$ results from the solution of the weak form

$$\int_\Omega \beta^* \left(\frac{dX}{dx}\right)^2 \beta^{p,\text{srb}}\, dx = \int_\Omega \beta^* f(x; s_p)\, dx, \tag{12}$$

for all $\beta^* \in \mathbb{R}$.

Obviously the solution $\beta^{p,\text{srb}}$ coincides with $S(s_p)$, a fact that validates the procedure.

2.1.2. Parametric reduced basis – prb – In this section we assume that the solution is now searched in the online stage according to

$$u(x; s) = \sum_{i=1}^{N} \beta_i^{p,\text{prb}} X_i(x) \cdot S_i(s). \tag{13}$$

In the present case, independently on the source location, the exact solution is obtained for $\beta_i^{p,\text{prb}} = 1, \forall s \in \mathcal{S}$ and $i = 1, \ldots, N$.

To check the procedure we consider again the reduced basis consisting of one single functional pair $X(x) \cdot S(s)$. We then proceed from the doubly-weak form (note that this is in sharp contrast to the standard weak form employed in the previous case)

$$\int_{\Omega\times\mathcal{S}} \beta^* S^2(s) \left(\frac{dX}{dx}\right)^2 \beta^{p,\text{prb}}\, dx\, ds = \int_{\Omega\times\mathcal{S}} \beta^* X(x) S(s) f(x; s_p)\, dx\, ds, \tag{14}$$

for all $\beta^* \in \mathbb{R}$, whose terms involving the s-coordinate can be integrated. By introducing the notation

$$\begin{cases} \alpha = \int_{\mathcal{S}} S^2(s)\, ds \\ \gamma = \int_{\mathcal{S}} S(s)\, ds \end{cases}, \tag{15}$$

Eq. (14) reduces to

$$\alpha \int_\Omega \beta^* \left(\frac{dX}{dx}\right)^2 \beta^{p,\text{prb}}\, dx = \gamma \int_\Omega \beta^* X(x) f(x; s_p)\, dx, \tag{16}$$

or, being $\alpha \neq 0$,

$$\int_\Omega \beta^* \left(\frac{dX}{dx}\right)^2 \beta^{p,\text{prb}}\, dx = \frac{\gamma}{\alpha} \int_\Omega \beta^* X(x) f(x; s_p)\, dx. \tag{17}$$

By comparing Eqs. (12) and (17) we obtain

$$\beta^{p,\text{prb}} = \frac{\gamma}{\alpha} \beta^{p,\text{srb}}, \tag{18}$$

DOI: 10.1002/nme

or, more explicitly

$$\beta^{p,\text{prb}} = \frac{\int_{\mathcal{S}} S(s)\, ds}{\int_{\mathcal{S}} S^2(s)\, ds}\, S(s_p), \tag{19}$$

that in general gives rise to $\beta^{p,\text{prb}} \neq 1$.

The above discussion allows us to conclude that there are two different routes for the usage of PGD parametric basis *à la* POD. The first one consist in particularizing first the value of the parameter and then projecting, while the second one consists in using the doubly weak form, without any parameter particularization. In Section 5 we compare and analyze results provided by both approaches.

3. SIMPLIFIED PHYSICS OF THE CUTTING PROCEDURE

We follow strictly the simplified physics of the cutting procedure assumed in [50]. In this reference, once contact between the surgical tool and the organ under consideration has been detected by a suitable contact algorithm (see [5] for instance, for a valid contact criterion in reduced model settings or, more adequately, [28] for an example using PGD), a criterion must be set in order to determine if a crack is produced or not, thus generating a new surface boundary in the domain, or not. Although complex physics occur, [11], the criterion established in [8] is followed here closely. This criterion has demonstrated to provide realistic enough results in haptic environments, even if it simplifies the actual physics taking place during surgery.

Since a scalpel produces a cut in the plane defined by its blade, the acting force is decomposed as indicated in Fig. 2:

$$\boldsymbol{F}_{\text{ext}} = \boldsymbol{F}_{\perp} + \boldsymbol{F}_{\parallel} = \boldsymbol{F}_{\perp} + \boldsymbol{F}_a + \boldsymbol{F}_n \tag{20}$$

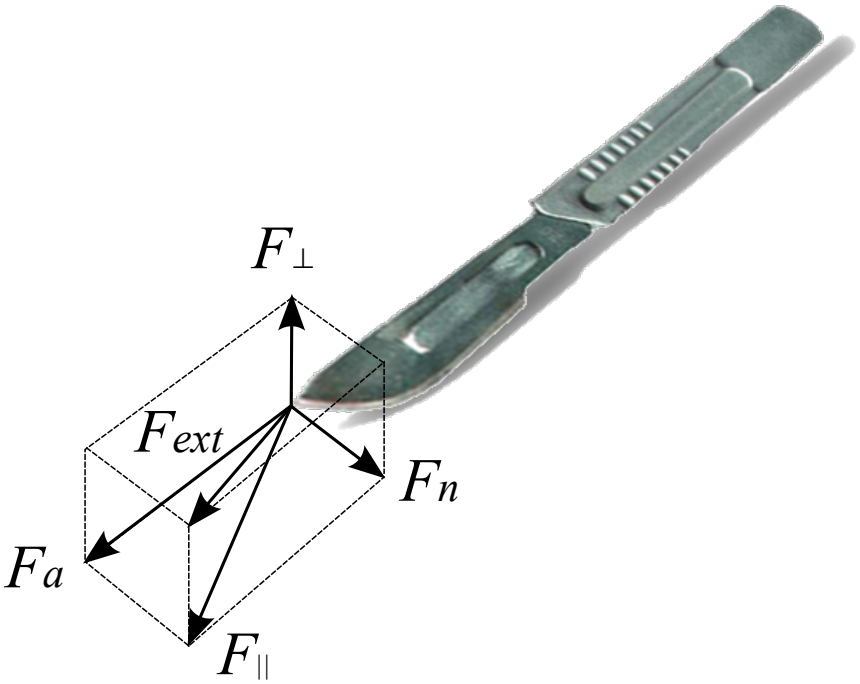

Figure 2. Force decomposition at the scalpel point of contact.

In the afore mentioned reference, a threshold value of the force F_{cut} is considered such that forces $\boldsymbol{F}_{\parallel}$ with modulus smaller than F_{cut} produce friction, but no cut. Once $\|\boldsymbol{F}_{\parallel}\|$ exceeds F_{cut}, the cut is

DOI: 10.1002/nme

produced and a discontinuity in the displacement field must be incorporated into the model. In this work a value $F_{\text{cut}} = 3N$ has been considered, in the absence of any experimental result.

In order to simplify the process and to make it simpler and (notably) faster, once the threshold value F_{cut} is reached, a whole finite element is then cut. No cut of length smaller the the typical element size h is considered. If the finite element mesh is dense enough, this limitation does not very much affect the results. Remember that the size of the global finite element mesh does not alter the number of degrees of freedom of the reduced model, as will be clear hereafter.

4. REAL-TIME CUTTING SIMULATION

The problem of constructing a computational vademecum [14] for the response of an organ to both a stab incision and the load transmitted by the contact of the surgical tool *could* be formulated as to determine the displacement at any point of the model, for any load position, for any force vector orientation and module, and for any path of the cut. As the reader will easily imagine, this "brute force" approach is out of reach due to both the complexity of the resulting formulation and the computer cost of the off-line phase of the method.

The approach followed here to solve this problem is to consider the solution $\boldsymbol{u}$ as given by a vademecum (like in [51] [52]) while there is no appearance of cuts in the on-line (real time) simulation. On the contrary, the simulation will abandon the vademecum once a cut is produced, and the model will be enriched on the fly accordingly. The vademecum at this stage will be considered a a reduced basis for representing the smooth global component of the deformation.

During this "cutting" phase of the simulation, the displacement field $\boldsymbol{u}$ will be simulated in a multiscale framework, as a continuous approximation enriched by a discontinuous, local field generated by the applied cuts:

$$\boldsymbol{u} = \boldsymbol{u}^{\text{cont}} + \boldsymbol{u}^{\text{disc}}. \tag{21}$$

Both contributions involve offline stages detailed below. Finally both will be combined online as detailed in the last part of the present section. While several possibilities arise to treat the discontinuous part of the displacement, such a [61], for instance, we have preferred to employ the so-called cracking node method [59]. It has been preferred since it provides naturally a discontinuous approximation that can be used with almost no modification also for visualization purposes. Details are given in subsequent sections.

4.1. Off-line stage for the continuous part of the approximation

The continuous part of the displacement field, $\boldsymbol{u}^{\text{cont}}$, will be simulated by taking the already computed PGD modes as basis functions. These basis functions are in fact global (Ritz-like) and parametric, since they depend on the position of the contact point between scalpel and organ. For a two-parameter approximation of the displacement, $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, \boldsymbol{s})$ it has been demonstrated that these modes coincide with the POD or SVD eigenfunctions, and therefore they are optimal in the sense that they incorporate the most of the energy of the system with the minimal number of degrees of freedom [42].

In Appendix A a detailed review of the computation of the sought PGD modes is included for completeness. This continuous part of the approximation will have the form

$$u_j^{\text{cont}}(\boldsymbol{x}, \boldsymbol{s}) = \sum_{k=1}^{n} \beta_j^k \cdot X_j^k(\boldsymbol{x}) \cdot Y_j^k(\boldsymbol{s}), \tag{22}$$

where the projection coefficients β_j^k will be determined on-line, as detailed in Section 4.3. It is also important to highlight that obtaining an expression similar to Eq. (22), POD techniques would need for a complete sampling of the s-parameter space. In other words, to obtain a snapshot for every possible nodal location of the load. This is something out of reach for the vast majority of real-life models.

DOI: 10.1002/nme

It is also noteworthy to highlight here that PGD methods are able, by means of asymptotic expansions, to provide a suitable approximation to non-linear displacement fields $\boldsymbol{u}^{\text{cont}}$ without the need for an update of global stiffness matrix, typical of classical POD-based methods [48] [12].

4.2. Offline stage for the discontinuous approximation

Extended finite element methods [22] have proved to constitute an appealing alternative for an efficient simulation of evolving cracks, such as cuts performed on soft living tissues [50] [34]. In essence, this technique consists in adding to the approximation new degrees of freedom representing the amplitude of a discontinuous field. One particular implementation of these techniques has received the name of *cracking node* method [59] and is particularly well suited for the goals here established.

Essentially, see Fig. 3, the method parametrizes the crack or cut by using nodally-centered cracks that extend up to the boundary of each element. Therefore, more than a single crack, the true geometry of the (possibly curved) crack is approximated by a collection of crack segments passing through the nodes.

Although the cracking node method could be seen as a poor approximation to the problem, the reader must keep in mind that in the type of applications pursued in this work the required level of accuracy is much lower than in usual engineering practice. Noteworthy, mechanical properties of living tissues often show standard deviations on the order of the mean values, so this method will not impose any special limitation on the accuracy of the results.

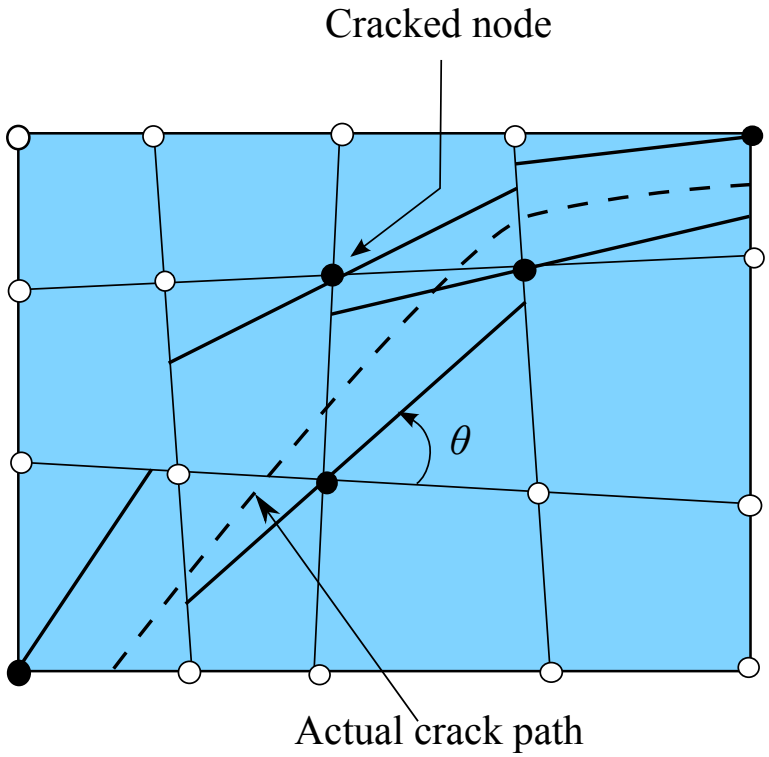

Figure 3. Sketch of the principles of the cracking node method. Adapted form [59].

Cracking-node techniques will be used to get $\boldsymbol{u}^{\text{disc}}$ through discontinuous enrichment functions $\tilde{H}_j$,

$$\boldsymbol{u}^{\text{disc}} = \boldsymbol{u}^{\text{XFEM}}(\boldsymbol{x},\theta) = \sum_{i\in\mathcal{S}} N^i(\boldsymbol{x})\tilde{H}^i(\boldsymbol{x},\theta)\boldsymbol{q}^i(t), \tag{23}$$

DOI: 10.1002/nme

where $N^i(\boldsymbol{x})$ denote the usual finite element shape functions, $\boldsymbol{q}^i(t)$ represent the nodal coefficients, that possibly depend on time for dynamical applications (only quasi-static examples are considered here, related to a pseudo-time, rather than time), and physically control the amplitude of the displacement discontinuity. $\mathcal{S}$ represents the set of nodes affected by the crack, and therefore with enriched degrees of freedom. Their associated shape functions, $N^i(\boldsymbol{x}) \cdot \tilde{H}^i(\boldsymbol{x}, \theta)$ are implemented with a parametric dependence on the angle of the crack towards the reference coordinates of the element, θ, with an eye towards time savings. However, θ will not be separated as a independent coordinate of the problem. Only the dependence of $\tilde{H}^i(\boldsymbol{x}, \theta)$ is highlighted. We refer the interested reader to the original reference [59] for more details of the implementation of the cracking node method.

It is worthy of noting that the use of the cracking-node method greatly simplifies the procedure introduced in our previous reference [50] for the on-line part of the simulation. In particular, it avoids the need to use an enrichment through the s-FEM method [25] all along the path of the X-FEM enriched model.

4.3. Online stage

During the online stage, real-time cutting is performed following interactively the path indicated by the user with the haptic device. Displacements produced by the interactive cutting are computed by projecting the solution onto a subspace spanned by the approximation functions (both the continuous and discontinuous parts) pre-computed in the offline stage. The parametric dependence of the global shape functions on the particular position of contact of the scalpel should be emphasized again here. Indeed, the computation time required at this stage must be compatible with the required haptic feedback rates (500-1000 Hz). To achieve this, an efficient method to compute the solution should be developed.

Both the PGD basis where to project $\boldsymbol{u}^{\text{cont}}$ and the discontinuous enrichment $\boldsymbol{u}^{\text{disc}}$ obtained in the off-line stage are used now as a reduced basis to project the solution $\boldsymbol{u}$. So to speak, PGD parametric basis functions are now employed *à la* POD to find a suitable projection, enriched with the discontinuous X-FEM field.

The (doubly-)weak form of the problem (see [51] [52] for details on the implementation in the hyperelastic framework), consists in finding the displacement $\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{s}) \in \mathcal{H}^1(\Omega) \times L^2(\bar{\Gamma})$ such that for all $\boldsymbol{u}^* \in \mathcal{H}^1_0(\Omega) \times L^2(\bar{\Gamma})$:

$$\int_{\bar{\Gamma}} \int_{\Omega} \boldsymbol{\nabla}_s \boldsymbol{u}^* : \boldsymbol{\sigma} \, d\Omega \, d\bar{\Gamma} = \int_{\bar{\Gamma}} \int_{\Gamma_t} \boldsymbol{u}^* \cdot \boldsymbol{t} \, d\Gamma \, d\bar{\Gamma}, \tag{24}$$

whereas in this case $\boldsymbol{u}$ is assumed to have the following form:

$$u_j(\boldsymbol{x}, \boldsymbol{s}, \theta) \approx \underbrace{\sum_{k=1}^{n} \beta_j^k \cdot X_j^k(\boldsymbol{x}) \cdot Y_j^k(\boldsymbol{s})}_{\boldsymbol{u}^{\text{cont}}} + \underbrace{\sum_{\ell=1}^{m} q_j^\ell \cdot N^\ell(\boldsymbol{x}) \cdot \tilde{H}^\ell(\boldsymbol{x}, \theta)}_{\boldsymbol{u}^{\text{disc}}}, \tag{25}$$

where β_j^k and q_j^ℓ, to be determined, can be considered as a sort of weighting parameters that allow balancing the already known expressions $X_j^k(\boldsymbol{x}) \cdot Y_j^k(\boldsymbol{s})$ and $N^\ell(\boldsymbol{x}) \cdot \tilde{H}^\ell(\boldsymbol{x}, \theta)$, i. e. the PGD approximation and its discontinuous enrichment. Continuity of the displacement field is ensured by the compact support of the typical finite element shape functions $N^\ell(\boldsymbol{x})$. The admissible variation of the displacement will thus be given by

$${u_j}^*(\boldsymbol{x}, \boldsymbol{s}, \theta) = \sum_{k=1}^{n} \beta_j^{k*} \cdot X_j^k(\boldsymbol{x}) \cdot Y_j^k(\boldsymbol{s}) + \sum_{\ell=1}^{m} q_j^{\ell*} \cdot N^\ell(\boldsymbol{x}) \cdot \tilde{H}^\ell(\boldsymbol{x}, \theta). \tag{26}$$

Remark 1

Note that in the case of the srb approach there is no need to employ a doubly weak form, since the basis no longer contains functions of load location ($\boldsymbol{s}$ coordinate). Therefore, the weak form of

DOI: 10.1002/nme

the problem is the standard one, i.e., find the displacement $\boldsymbol{u}(\boldsymbol{x},\boldsymbol{s}) \in \mathcal{H}^1(\Omega \times \bar{\Gamma})$ such that for all $\boldsymbol{u}^* \in \mathcal{H}_0^1$:

$$\int_\Omega \boldsymbol{\nabla}_s \boldsymbol{u}^* : \boldsymbol{\sigma}\, d\Omega = \int_{\Gamma_t} \boldsymbol{u}^* \cdot \boldsymbol{t}\, d\Gamma. \tag{27}$$

By substituting Eqs. (25) and (26) into the *doubly* weak form of the problem, see Eq. (32) in Ap. A, a discrete expression for getting β_j^k, and q_j^ℓ can be established as:

$$\begin{pmatrix} \boldsymbol{K}^{\beta\beta} & \boldsymbol{K}^{\beta q_1} & \boldsymbol{K}^{\beta q_2} & \dots & \boldsymbol{K}^{\beta q_m} \\ \boldsymbol{K}^{q_1\beta} & \boldsymbol{K}^{q_1 q_1} & \boldsymbol{K}^{q_1 q_2} & \dots & \boldsymbol{K}^{q_1 q_m} \\ \boldsymbol{K}^{q_2\beta} & \boldsymbol{K}^{q_2 q_1} & \boldsymbol{K}^{q_2 q_2} & \dots & \boldsymbol{K}^{q_2 q_m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{K}^{q_m\beta} & \boldsymbol{K}^{q_m q_1} & \boldsymbol{K}^{q_m q_2} & \dots & \boldsymbol{K}^{q_m q_m} \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{q}_1 \\ \boldsymbol{q}_2 \\ \vdots \\ \boldsymbol{q}_m \end{pmatrix} = \begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{0} \\ \boldsymbol{0} \\ \vdots \\ \boldsymbol{0} \end{pmatrix}. \tag{28}$$

Remark 2
Note that the angle of crack, θ, is not considered as a parameter in the formulation. This means that no integral in θ is performed on the weak form of the problem. Instead, the value of θ imposed by the blade of the scalpel is particularized in $\tilde{H}^\ell(\boldsymbol{x},\theta)$ so as to provide a closed-form expression $\tilde{H}^\ell(\boldsymbol{x})$.

Note also that, in an interactive simulation, the number m of nodes belonging to the *active* set $\mathcal{S}$, affected by the crack, increases as the surgeon proceeds. Therefore, the size of the stiffness matrix is also increasing during the interactive simulation. However, since the modes controlling the continuous part of the displacement of the organ are those related to the PGD solution, and therefore very limited in number, the size of the stiffness matrix does not increase drastically. In the numerical experiments performed so far, this size did not prevented the algorithm from running under real-time constraints, as will be demonstrated, even for very rude Matlab code prototypes.

5. NUMERICAL RESULTS: CUTTING THE CORNEA

In this section an example of corneal surgery is studied [55]. In particular, astigmatism surgery by means of radial keratotomy is considered. This type of surgery consists of the performance of radial incisions in the corneal tissue with a diamond knife. The idea is to perform a change of curvature in the cornea, able to reduce the lack of sphericity of the cornea and to potentially eliminate the defects associated to myopia or astigmatism.

The main objective of this numerical example is to reproduce the cut on the cornea with reasonable accuracy, compatible with the physical sensation obtained by a surgeon. To that end, a finite element model of the cornea, developed by prof. B. Calvo and coworkers [1] [41] is employed.

5.1. Finite element model of the cornea

The cornea was meshed using trilinear hexahedral elements. It consisted of 8514 nodes and 6840 elements. The mesh is shown in Figure 4 in two views. The cornea is clamped at its base, giving a dome-like geometry.

Corneal tissue is assumed to be hyperelastic, in accordance to the vast majority of the literature [1] [41] [52]. In this case, for simplicity of the exposition, a Kirchhoff-Saint Venant behaviour was assumed. Although this model is well known to present some instabilities in compression is among the most sophisticated ones employed in real-time simulation. More sophisticated material behaviours can also be efficiently considered, as in [48], where a two-families of fibers reinforced hyperelasticity model was successfully employed, or as in [52], where a neo-Hookean material was employed under very astringent real-time constraints. The material properties of the cornea were assumed to be $E = 2MPa$ and $\nu = 0.48$ [1].

The Kirchhoff-Saint Venant model is characterized by the energy function given by

$$\Psi = \frac{\lambda}{2}(\mathrm{tr}(\boldsymbol{E}))^2 + \mu \boldsymbol{E} : \boldsymbol{E} \tag{29}$$

DOI: 10.1002/nme

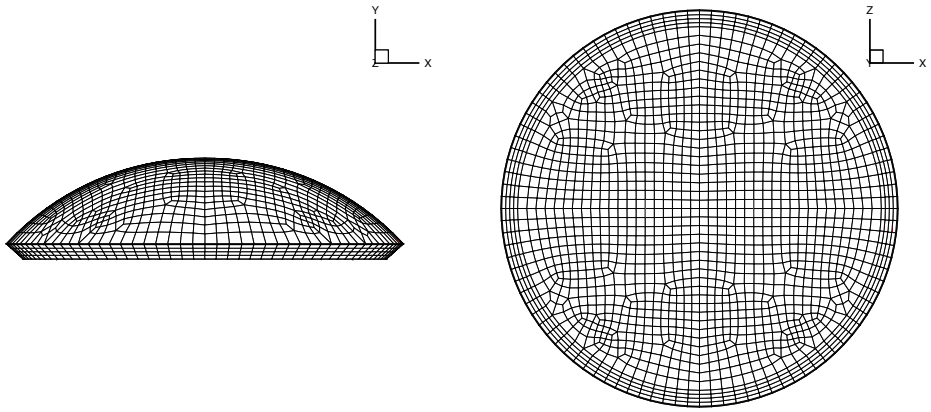

Figure 4. Geometry of the finite element model for the human cornea [1].

where λ and μ are Lame's constants. The Green-Lagrange strain tensor, $\boldsymbol{E}$, has the form

$$\boldsymbol{E} = \frac{1}{2}(\boldsymbol{F}^T\boldsymbol{F} - 1) = \gamma_l(\boldsymbol{u}) + \gamma_{nl}(\boldsymbol{u}, \boldsymbol{u}) \tag{30}$$

where $\boldsymbol{F} = \boldsymbol{\nabla u} + \boldsymbol{I}$ is the gradient of deformation tensor. The second Piola-Kirchhoff stress tensor can be obtained by

$$\boldsymbol{S} = \frac{\partial \Psi(\boldsymbol{E})}{\partial \boldsymbol{E}} = \mathbf{C} : \boldsymbol{E} \tag{31}$$

in which $\mathbf{C}$ is the fourth-order constitutive (elastic) tensor.

5.2. Radial incisions on the cornea

Under the before commented assumptions, the cornea model has been subjected to different patterns of radial, spoke-shaped incisions. See Fig. 5 for different snapshots of the cutting procedure.

PGD modes employed as a Ritz basis for the problem are shown in Fig. 6. Only six modes were enough to reproduce the results shown before. The portion of the cornea amenable to suffer cut was considered as a strip of 3×18 nodes around the zone where the incision is made. Noteworthy, only six modes were enough to reproduce the cutting procedure with reasonable accuracy. In general, obtained results produce a very realistic sensation, both from their appearance (the method very much improves the quality of the author's previous method, see [50]) and from the haptic response perceived (previous results were able to run only under visual constraints, some 25 Hz). L^2-error norms were computed by taking a full finite element simulation as a reference solution. Those errors are reported in Fig. 7, where a comparison is made of the relative errors for both approaches described in Section 2.1. Noteworthy, as expected, the *space reduced basis* approach rendered better results. Nevertheless, in all our simulations the reported error versus a full finite element simulation stayed below 7×10^{-3}, which is far more than usual in this type of simulations.

5.3. Timings

The results presented in this section have been obtained with a 64-bit laptop, running Matlab 2014b with an i5 processor running at 2.50 GHz, 4Gb RAM memory. Despite the use of crude Matlab code prototypes, the examples of the cornea ran always faster than 1 kHz, which is enough for visual as well as haptic real-time requirements. The summary of the tests performed, which involved different cut lengths, are shown in Fig. 8. It is worthy of noting that the results seem to be almost independent of the length of the cut, although we assume that there should be some limit.

Prepared using nmeauth.cls
DOI: 10.1002/nme

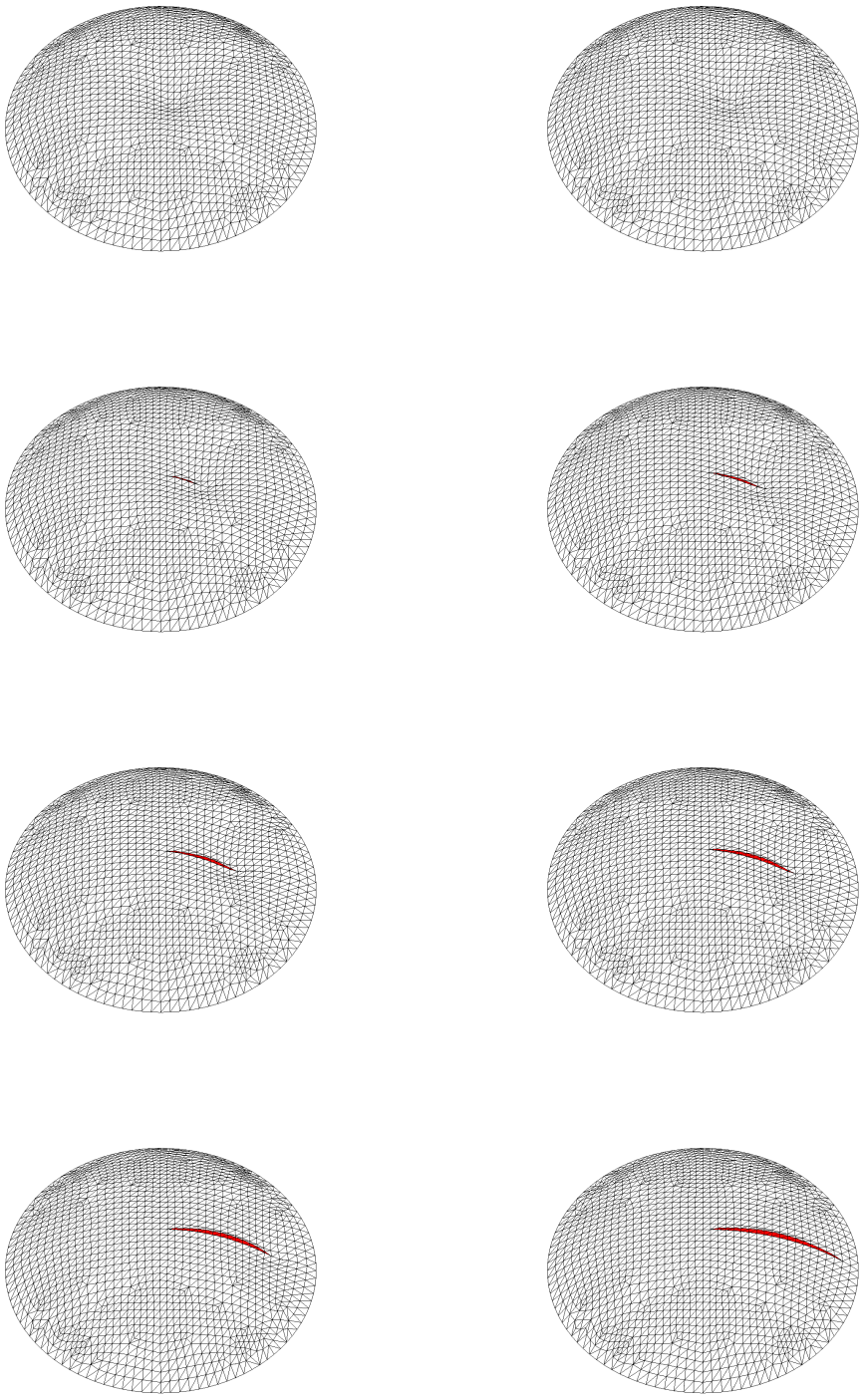

Figure 5. Different snapshots of the radial cutting procedure.

6. CONCLUSIONS

In this work a new method for simulation of surgical cutting under real-time (haptic) constraints has been developed. The method is based on two main ingredients. On one side, the computation

Prepared using nmeauth.cls
DOI: 10.1002/nme

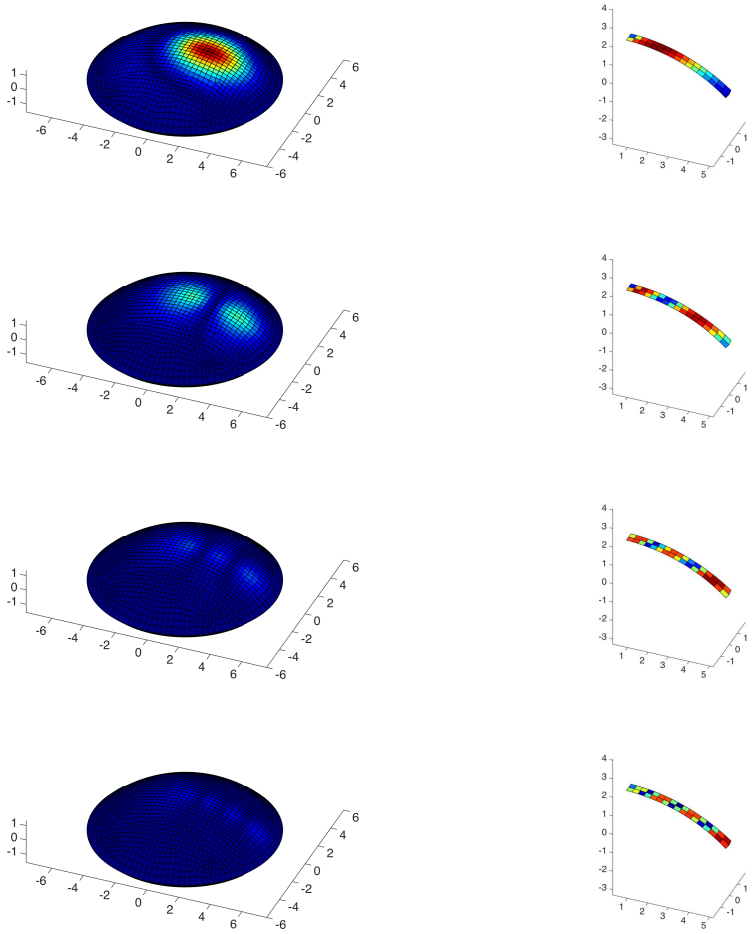

Figure 6. First four PGD modes employed in the simulation of radial incision. Left column, spatial modes. Right column, loading modes.

of a computational vademecum, covering the response of the organ (in this case, the human cornea) to any possible contact position with the scalpel. This vademecum is then enriched, on line, with discontinuous (X-FEM like) shape functions. This enrichment can not be reasonably covered in a single vademecum, since the results for any possible position, orientation and length of a cut can not be easily compressed. In other words, the problem is not *separable* in the form given by Eq. 34.

If compared to the author's previous work [50], this work introduces two main additional advantages. One is the employ, as mentioned before, of multi-dimensional PGD results *à la* POD, i.e., the problem is solved by a Galerkin projection onto a set of multidimensional basis. These basis are obtained by PGD techniques and are much richer than those obtained by POD (like in [50]) and do not need the computation of snapshots, solutions of complete problems under different loading conditions.

On the other side, the second main ingredient is the employ of the *cracking node* method [59]. This particular version of the X-FEM technique greatly simplifies the treatment of the discontinuous enrichments, therefore allowing to great CPU savings by simply storing in memory many of the matrices needed to perform the simulation, as explained in Section 4.2.

The cracking node method is inaccurate in the sense that it does not provide a continuous description of the crack geometry. It is not the purpose of the paper to discuss on a method that has been previously validated in the literature. Our previous approach [51], that indeed considered

DOI: 10.1002/nme

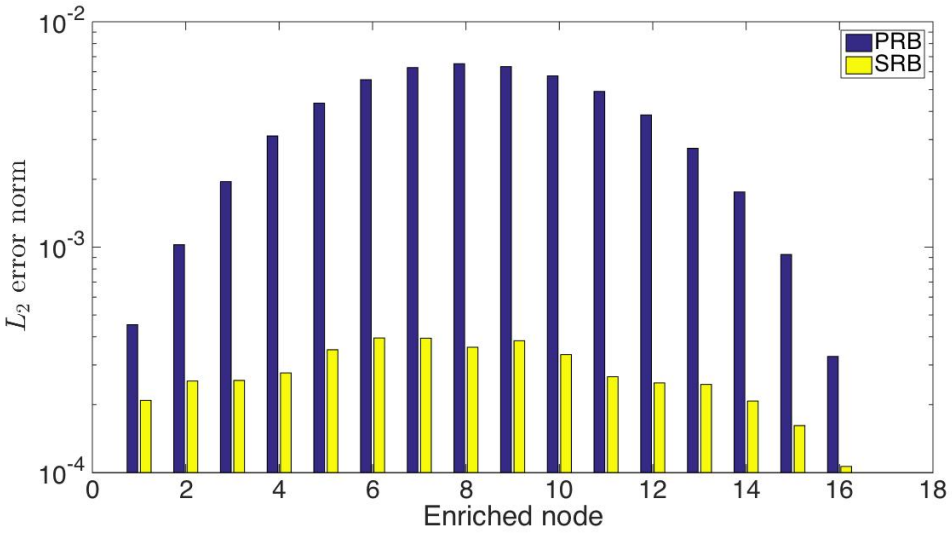

Figure 7. L^2-norm error produced in the cutting simulation as the scalpel advanced. Errors were measured in L^2-norm with respect to a full finite element solution of the same cutting procedure. Both approaches, namely prb and srb, are represented. In abscissae, a measure of the length of the cut is presented through the number of nodes belonging to the cut zone.

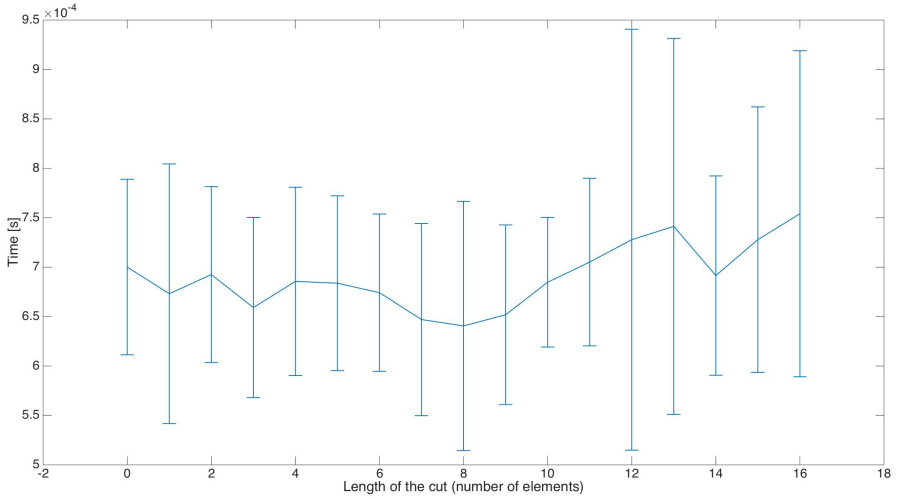

Figure 8. Statistics of the computing time employed for different cut length simulations. Note that every example ran below 1 ms.

a full XFEM description of the cut, ran at some 25 Hz, enough for visual rendering, but not for haptics. Thus, it should be highlighted that in this field the true limitation is not accuracy itself (even if it is undoubtedly important), but one's ability to simulate at such impressive speeds.

On the other hand, but strictly related to this is the need of using ROM. As demonstrated in our previous works, there seems to be no other way to be able to reproduce large strains. Only some works by K. Miller [47] [36], employing element-by-element explicit finite elements, are able to accomplish it. But in this case, long simulations become instable, as it is well known. To the best of our knowledge, the use of ROM is the only way to obtain stable dynamical simulations for large

DOI: 10.1002/nme

strain hyperelastic models. And the cracking node is the best way to describe cuts. There seem not to be any other alternative in the literature, at this moment.

In sum, the successful combination of PGD basis (employed in a POD framework) and the cracking node method allows for very efficient real-time simulations. This method has been tested on a GeoMagic SensAble haptic peripheral [27], providing excellent results.

A. OFFLINE COMPUTATION OF THE PGD MODES FOR THE CONTINUOUS PART OF THE APPROXIMAITON

The obtaining of the PGD modes for the continuous part of the displacement field is briefly reviewed here. For a more detailed insight, the interested reader is referred to some of the authors' previous works [51] [52], with a special emphasis on the treatment of the non-linear character of the constitutive equations for most living soft tissues without the need for updates in the global stiffness matrix. Consider the simplest case of such a displacement field, defined as a parametric solution $\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{s})$, in which $\boldsymbol{x}$ represents the physical coordinates of a point in the organ of interest, Ω, and $\boldsymbol{s} \in \bar{\Gamma} \subset \Gamma_t \subset \Gamma = \partial\Omega$ represents the point of contact between surgical tool and organ. Here, $\Gamma = \Gamma_u \cup \Gamma_t$ represents the boundary of the organ, as a union of the essential (Dirichlet) and natural (Neumann) parts of the boundary, respectively; $\bar{\Gamma} \subset \Gamma_t$ is the boundary which is accessible to the surgeon.

In such a setting, the weak form of the equilibrium equations (balance of linear momentum) is considered, omitting inertia terms for the sake of simplicity. The dynamic case was considered in [29] or [9]. The problem can thus be thought of as finding the displacement $\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{s}) \in \mathcal{H}^1(\Omega \times \bar{\Gamma})$ such that for all $\boldsymbol{u}^* \in \mathcal{H}^1_0$:

$$\int_{\bar{\Gamma}} \int_{\Omega} \boldsymbol{\nabla}_s \boldsymbol{u}^* : \boldsymbol{\sigma} \, d\Omega \, d\bar{\Gamma} = \int_{\bar{\Gamma}} \int_{\Gamma_t} \boldsymbol{u}^* \cdot \boldsymbol{t} \, d\Gamma \, d\bar{\Gamma}. \tag{32}$$

The load $\boldsymbol{t}$ can be approximated as a truncated series of separable functions in the spirit of the PGD method:

$$t_j \approx \sum_{i=1}^{m} f_j^i(\boldsymbol{x}) \cdot g_j^i(\boldsymbol{s}) \tag{33}$$

where m represents the order of truncation and f_j^i, g_j^i represent the j-th component of vectorial functions in space and boundary position, respectively.

The PGD approach to the problem is characterized by the construction, in an iterative way, of an approximation to the solution in the form of a finite sum of separable functions. If convergence is assumed at iteration n of this procedure,

$$u_j^n(\boldsymbol{x}, \boldsymbol{s}) = \sum_{k=1}^{n} X_j^k(\boldsymbol{x}) \cdot Y_j^k(\boldsymbol{s}), \tag{34}$$

where the term u_j refers to the j-th component of the displacement vector, $j = 1, 2, 3$ and functions X_j^k and Y_j^k represent the j-th component of the separated functions used to approximate the unknown field, obtained in previous iterations of the PGD algorithm.

An improvement of this approximation, the $(n+1)$-th term, will look like

$$u_j^{n+1}(\boldsymbol{x}, \boldsymbol{s}) = u_j^n(\boldsymbol{x}, \boldsymbol{s}) + R_j(\boldsymbol{x}) \cdot S_j(\boldsymbol{s}), \tag{35}$$

where $\boldsymbol{R}(\boldsymbol{x})$ and $\boldsymbol{S}(\boldsymbol{s})$ are the sought functions that improve the approximation.

In this framework, the admissible variation of the displacement will be given by

$$u_j^*(\boldsymbol{x}, \boldsymbol{s}) = R_j^*(\boldsymbol{x}) \cdot S_j(\boldsymbol{s}) + R_j(\boldsymbol{x}) \cdot S_j^*(\boldsymbol{s}). \tag{36}$$

At this point, several options are available to determine the new pair of functions $\boldsymbol{R}$ and $\boldsymbol{S}$. The most frequently used in the PGD framework, due to both its easy of implementation and good

DOI: 10.1002/nme

convergence properties, is a fixed-point algorithm in which functions $\boldsymbol{R}$ and $\boldsymbol{S}$ are sought alternately in an iterative way. The implementation of this algorithm is briefly described below.

First, $\boldsymbol{S}(\boldsymbol{s})$ will be computed assuming that $\boldsymbol{R}(\boldsymbol{x})$ is known. In this case, following standard assumptions of variational calculus,

$$u_j^*(\boldsymbol{x},\boldsymbol{s}) = R_j(\boldsymbol{x}) \cdot S_j^*(\boldsymbol{s}), \tag{37}$$

or, equivalently, $\boldsymbol{u}^*(\boldsymbol{x},\boldsymbol{s}) = \boldsymbol{R} \circ \boldsymbol{S}^*$, where the symbol "$\circ$" denotes the so-called entry-wise, Hadamard or Schur multiplication for vectors. Once substituted into Eq. (32) and after some operation, the following expression is obtained:

$$\int_{\bar{\Gamma}} \int_{\Omega} (\boldsymbol{\nabla}_s \boldsymbol{R} \circ \boldsymbol{S}^*) : \mathbf{C} : (\boldsymbol{\nabla}_s \boldsymbol{R} \circ \boldsymbol{S})\, d\Omega\, d\bar{\Gamma} = \int_{\bar{\Gamma}} \int_{\Gamma_t} (\boldsymbol{R} \circ \boldsymbol{S}^*) \cdot \left(\sum_{k=1}^{m} \boldsymbol{f}^k \circ \boldsymbol{g}^k \right) d\Gamma\, d\bar{\Gamma} - \int_{\bar{\Gamma}} \int_{\Omega} (\boldsymbol{\nabla}_s \boldsymbol{R} \circ \boldsymbol{S}^*) \cdot \mathcal{R}^n\, d\Omega\, d\bar{\Gamma}$$

with $\mathcal{R}^n = \mathbf{C} : \boldsymbol{\nabla}_s \boldsymbol{u}^n$. All the terms depending on $\boldsymbol{x}$ are known and hence all integrals over Ω and Γ_t (the support of the punctual load) can be computed to derive an equation to obtain $\boldsymbol{S}(\boldsymbol{s})$.

Equivalently, in the case in which $\boldsymbol{R}(\boldsymbol{x})$ must be computed assuming that $\boldsymbol{S}(\boldsymbol{s})$ is known, the admissible variation can be written as:

$$u_j^*(\boldsymbol{x},\boldsymbol{s}) = R_j^*(\boldsymbol{x}) \cdot S_j(\boldsymbol{s}), \tag{38}$$

which, once substituted into Eq. (32), gives

$$\int_{\bar{\Gamma}} \int_{\Omega} \boldsymbol{\nabla}_s (\boldsymbol{R}^* \circ \boldsymbol{S}) : \mathbf{C} : \boldsymbol{\nabla}_s \left(\sum_{k=1}^{n} \boldsymbol{X}^k \circ \boldsymbol{Y}^k + \boldsymbol{R} \circ \boldsymbol{S} \right) d\Omega d\bar{\Gamma} = \int_{\bar{\Gamma}} \int_{\Gamma_t} (\boldsymbol{R}^* \circ \boldsymbol{S}) \cdot \left(\sum_{k=1}^{m} \boldsymbol{f}^k \circ \boldsymbol{g}^k \right) d\Gamma d\bar{\Gamma}.$$

In this case all the terms depending on the load position $\boldsymbol{s}$ can be integrated over $\bar{\Gamma}$, leading to a generalized elastic problem to compute the function $\boldsymbol{R}(\boldsymbol{x})$.

This simple algorithm renders, in general, excellent convergence properties (see [17] and references therein).

REFERENCES

1. V. Alastrue, B. Calvo, E. Pena, and M. Doblare. Biomechanical modeling of refractive corneal surgery. *Journal of Biomechanical Engineering-Trasactions of the ASME*, 128:150–160, 2006.
2. Iciar Alfaro, David Gonzalez, Felipe Bordeu, Adrien Leygue, Amine Ammar, Elias Cueto, and Francisco Chinesta. Real-time in silico experiments on gene regulatory networks and surgery simulation on handheld devices. *Journal of Computational Surgery*, 1(1):2194–3990, 2014.
3. Iciar Alfaro, David Gonzalez, Sergio Zlotnik, Pedro Diez, Elias Cueto, and Francisco Chinesta. An error estimator for real-time simulators based on model order reduction. *Advanced Modeling and Simulation in Engineering Sciences*, 2(1):30, 2015.
4. A. Ammar, E. Cueto, and F. Chinesta. Reduction of the chemical master equation for gene regulatory networks using proper generalized decompositions. *International Journal for Numerical Methods in Biomedical Engineering*, in press, 2012.
5. Jernej Barbič and Doug James. Time-critical distributed contact for 6-DoF haptic rendering of adaptively sampled reduced deformable models. In Metaxas, D and Popovic, J, editor, *SYMPOSIUM ON COMPUTER ANIMATION 2007: ACM SIGGRAPH/ EUROGRAPHICS SYMPOSIUM PROCEEDINGS*, pages 171–180, 1515 BROADWAY, NEW YORK, NY 10036-9998 USA, 2007. ACM SIGGRAPH; Eurog Assoc, ASSOC COMPUTING MACHINERY. Symposium on Computer Animation, San Diego, CA, AUG 03-04, 2007.
6. Jernej Barbič and Doug L. James. Real-time subspace integration for St. Venant-Kirchhoff deformable models. *ACM Transactions on Graphics (SIGGRAPH 2005)*, 24(3):982–990, August 2005.
7. C. Bernoulli. *Vademecum des Mechanikers*. Cotta, 1836.

DOI: 10.1002/nme

8. D. Bielser and M.H. Gross. Interactive simulation of surgical cuts. In *Computer Graphics and Applications, 2000. Proceedings. The Eighth Pacific Conference on*, pages 116–442, 2000.
9. L. Boucinha, A. Gravouil, and A. Ammar. Space-time proper generalized decompositions for the resolution of transient elastodynamic models. *Computer Methods in Applied Mechanics and Engineering*, 255(0):67 – 88, 2013.
10. M. Bro-Nielsen and S. Cotin. Real-time volumetric deformable models for surgery simulation using finite elements and condensation. *Computer Graphics Forum*, 15(3):57–66, 1996.
11. T. Chanthasopeephan, J.P. Desai, and A.C.W. Lau. Modeling soft-tissue deformation prior to cutting for surgical simulation: Finite element analysis and study of cutting parameters. *Biomedical Engineering, IEEE Transactions on*, 54(3):349–359, March 2007.
12. Saifon Chaturantabut and Danny C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, 32:2737–2764, September 2010.
13. F. Chinesta, A. Ammar, and E. Cueto. Recent advances in the use of the Proper Generalized Decomposition for solving multidimensional models. *Archives of Computational Methods in Engineering*, 17(4):327–350, 2010.
14. F. Chinesta, A. Leygue, F. Bordeu, J.V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, and A. Huerta. PGD-Based Computational Vademecum for Efficient Design, Optimization and Control. *Archives of Computational Methods in Engineering*, 20(1):31–59, 2013.
15. Francisco Chinesta and Elias Cueto. *PGD-Based Modeling of Materials, Structures and Processes*. Springer International Publishing Switzerland, 2014.
16. Francisco Chinesta, Roland Keunings, and Adrien Leygue. *The Proper Generalized Decomposition for Advanced Numerical Simulations*. Springer International Publishing Switzerland, 2014.
17. Francisco Chinesta, Pierre Ladeveze, and Elias Cueto. A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18:395–404, 2011.
18. Stéphane Cotin, Hervé Delingette, and Nicholas Ayache. Real-time elastic deformations of soft tissues for surgery simulation. In Hans Hagen, editor, *IEEE Transactions on Visualization and Computer Graphics*, volume 5 (1), pages 62–73. IEEE Computer Society, 1999.
19. Hadrien Courtecuisse, Jérémie Allard, Pierre Kerfriden, Stéphane P.A. Bordas, Stéphane Cotin, and Christian Duriez. Real-time simulation of contact and cutting of heterogeneous soft-tissues. *Medical Image Analysis*, 18(2):394 – 410, 2014.
20. Hadrien Courtecuisse, Hoeryong Jung, Jérémie Allard, Christian Duriez, Doo Yong Lee, and Stéphane Cotin. Gpu-based real-time soft tissue deformation with cutting and haptic feedback. *Progress in Biophysics and Molecular Biology*, 103(2-3):159 – 168, 2010. Special Issue on Biomechanical Modelling of Soft Tissue Motion.
21. Elias Cueto and Francisco Chinesta. Real time simulation for computational surgery: a review. *Advanced Modeling and Simulation in Engineering Sciences*, 1(1):11, 2014.
22. C Daux, N Moes, J Dolbow, N Sukumar, and T Belytschko. Arbitrary branched and intersecting cracks with the extended finite element method. *International Journal for Numerical Methods in Engineering*, 48(12):1741–1760, AUG 30 2000.
23. H. Delingette and N. Ayache. Soft tissue modeling for surgery simulation. In N. Ayache, editor, *Computational Models for the Human Body*, Handbook of Numerical Analysis (Ph. Ciarlet, Ed.), pages 453–550. Elsevier, 2004.
24. Herve Delingette and Nicholas Ayache. Hepatic surgery simulation. *Communications of the ACM*, 48:31–36, February 2005.
25. J Fish. The s-version of the finite-element method. *Computers and Structures*, 43(3):539–547, 1992.
26. M. Garbey, B. L. Bass, and S. Berceli. Multiscale mechanobiology modeling for surgery assessment. *ACTA MECHANICA SINICA*, 28(4):1186–1202, AUG 2012.
27. Geomagic. *OpenHaptics Toolkit*. 3D systems - Geomagic solutions, 430 Davis Drive, Suite 300 Morrisville, NC 27560 USA, 2013.
28. David González, Icíar Alfaro, Carlos Quesada, Elías Cueto, and Francisco Chinesta. Computational vademecums for the real-time simulation of haptic collision between nonlinear solids. *Computer Methods in Applied Mechanics and Engineering*, 283(0):210 – 223, 2015.
29. David Gonzalez, Elias Cueto, and Francisco Chinesta. Real-time direct integration of reduced solid dynamics equations. *Internatinal Journal for Numerical Methods in Engineering*, 99(9):633–653, 2014.
30. F. El Halabi, D. González, A. Chico, and M. Doblaré. {FE2} multiscale in linear elasticity based on parametrized microscale models using proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 257:183 – 202, 2013.
31. Wang He-xiang, Hao Ai-min, Li duo, and Liu Xue-mei. Real-time cutting method for soft tissue based on tled algorithm. In *Computer Engineering and Technology (ICCET), 2010 2nd International Conference on*, volume 3, pages V3–393–V3–396, April 2010.
32. Ch. Heyberger, P.-A. Boucard, and D. Neron. A rational strategy for the resolution of parametrized problems in the {PGD} framework. *Computer Methods in Applied Mechanics and Engineering*, 259(0):40 – 49, 2013.
33. S. R. Idelsohn and R. Cardona. A reduction method for nonlinear structural dynamics analysis. *Computer Methods in Applied Mechanics and Engineering*, 49:253–279, 1985.
34. Lenka Jeřábková and Torsten Kuhlen. Stable cutting of deformable objects in virtual environments using xfem. *IEEE Comput. Graph. Appl.*, 29(2):61–71, March 2009.
35. Xia Jin, Grand Roman Joldes, Karol Miller, King H. Yang, and Adam Wittek. Meshless algorithm for soft tissue cutting in surgical simulation. *Computer Methods in Biomechanics and Biomedical Engineering*, 17(7):800–811, 2014. PMID: 22974246.
36. Grand Roman Joldes, Adam Wittek, and Karol Miller. Real-time nonlinear finite element computations on {GPU} – application to neurosurgical simulation. *Computer Methods in Applied Mechanics and Engineering*, 199(49–52):3305 – 3314, 2010.
37. K. Karhunen. Uber lineare methoden in der wahrscheinlichkeitsrechnung. *Ann. Acad. Sci. Fennicae, ser. Al. Math. Phys.*, 37, 1946.

DOI: 10.1002/nme

38. P. Krysl, S. Lall, and J.E. Marsden. Dimensional model reduction in non-linear finite element dynamics of solids and structures. *Int. J. Numer. Meth. in Engng.*, 51:479–504, 2001.
39. P. Ladeveze, J.-C. Passieux, and D. Neron. The latin multiscale computational method and the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199(21-22):1287 – 1296, 2010.
40. Sanjay Lall, Petr Krysl, and Jerrold E Marsden. Structure-preserving model reduction for mechanical systems. *Physica D: Nonlinear Phenomena*, 184(1-4):304–318, 2003.
41. Elena Lanchares, Begoña Calvo, José A. Cristóbal, and Manuel Doblaré. Finite element simulation of arcuates for astigmatism correction. *Journal of Biomechanics*, 41(4):797 – 805, 2008.
42. C. Le Bris, T. Lelièvre, and Y. Maday. Results and questions on a nonlinear approximation approach for solving high-dimensional partial differential equations. *Constructive Approximation*, 30:621–651, 2009. 10.1007/s00365-009-9071-1.
43. M. M. Loève. *Probability theory*. The University Series in Higher Mathematics, 3rd ed. Van Nostrand, Princeton, NJ, 1963.
44. E. N. Lorenz. *Empirical Orthogonal Functions and Statistical Weather Prediction*. MIT, Departement of Meteorology, Scientific Report Number 1, Statistical Forecasting Project, 1956.
45. U. Meier, O. Lopez, C. Monserrat, M. C. Juan, and M. Alcaniz. Real-time deformable models for surgery simulation: a survey. *Computer Methods and Programs in Biomedicine*, 77(3):183–197, 2005.
46. Andres Mena, David Bel, Iciar Alfaro, David Gonzalez, Elias Cueto, and Francisco Chinesta. Towards a pancreatic surgery simulator based on model order reduction. *Advanced Modeling and Simulation in Engineering Sciences*, 2(1):31, 2015.
47. Karol Miller, Grand Joldes, Dane Lance, and Adam Wittek. Total lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation. *Communications in Numerical Methods in Engineering*, 23(2):121–134, 2007.
48. S. Niroomandi, I. Alfaro, E. Cueto, and F. Chinesta. Real-time deformable models of non-linear tissues by model reduction techniques. *Computer Methods and Programs in Biomedicine*, 91(3):223 – 231, 2008.
49. S. Niroomandi, I. Alfaro, E. Cueto, and F. Chinesta. Accounting for large deformations in real-time simulations of soft tissues based on reduced-order models. *Computer Methods and Programs in Biomedicine*, 105(1):1–12, 2012.
50. S. Niroomandi, I. Alfaro, D. Gonzalez, E. Cueto, and F. Chinesta. Real-time simulation of surgery by reduced-order modeling and x-fem techniques. *International Journal for Numerical Methods in Biomedical Engineering*, 28(5):574–588, 2012.
51. S. Niroomandi, D. González, I. Alfaro, F. Bordeu, A. Leygue, E. Cueto, and F. Chinesta. Real-time simulation of biological soft tissues: a PGD approach. *International Journal for Numerical Methods in Biomedical Engineering*, 29(5):586–600, 2013.
52. S. Niroomandi, D. Gonzalez, I. Alfaro, E. Cueto, and F. Chinesta. Model order reduction in hyperelasticity: a proper generalized decomposition approach. *International Journal for Numerical Methods in Engineering*, 96(3):129–149, 2013.
53. Siamak Niroomandi, Iciar Alfaro, Elias Cueto, and Francisco Chinesta. Model order reduction for hyperelastic materials. *International Journal for Numerical Methods in Engineering*, 81(9):1180–1206, 2010.
54. Anthony Nouy. A priori model reduction through Proper Generalized Decomposition for solving time-dependent partial differential equations. *COMPUTER METHODS IN APPLIED MECHANICS AND ENGINEERING*, 199(23-24):1603–1626, 2010.
55. Anna Pandolfi and Gerhard A. Holzapfel. Three-dimensional modeling and computational analysis of the human cornea considering distributed collagen fibril orientations. *Journal of Biomechanical Engineering*, 130(6):061006–061006, 10 2008.
56. E. Pruliere, F. Chinesta, and A. Ammar. On the deterministic solution of multidimensional parametric models using the Proper Generalized Decomposition. *Mathematics and Computers in Simulation*, 81(4):791–810, DEC 2010.
57. A. Radermacher and S. Reese. Proper orthogonal decomposition-based model reduction for nonlinear biomechanical analysis. *International Journal of Materials Engineering Innovation*, 4(4):149–165, 2013.
58. Sebastien Schmerber, Arnaud Attye, Ihab Atallah, Cedric Mendoza, and Alexandre Karkas. Normal temporal bone, ct. In StilianosE. Kountakis, editor, *Encyclopedia of Otolaryngology, Head and Neck Surgery*, pages 1876–1886. Springer Berlin Heidelberg, 2013.
59. Jeong-Hoon Song and Ted Belytschko. Cracking node method for dynamic fracture with finite elements. *International Journal for Numerical Methods in Engineering*, 77(3):360–385, 2009.
60. Z.A. Taylor, M. Cheng, and S. Ourselin. High-speed nonlinear finite element analysis for surgical simulation using graphics processing units. *Medical Imaging, IEEE Transactions on*, 27(5):650 –663, may 2008.
61. G.J. van Zwieten, E.H. van Brummelen, K.G. van der Zee, M.A. Gutiérrez, and R.F. Hanssen. Discontinuities without discontinuity: The weakly-enforced slip method. *Computer Methods in Applied Mechanics and Engineering*, 271:144 – 166, 2014.
62. Matthieu Vitse, David Neron, and Pierre-Alain Boucard. Virtual charts of solutions for parametrized nonlinear equations. *COMPUTATIONAL MECHANICS*, 54(6):1529–1539, DEC 2014.

DOI: 10.1002/nme