Controlling the error on target motion through real-time mesh adaptation: applications to Deep Brain Stimulation

Huu Phuoc Bui, Satyendra Tomar, Hadrien Courtecuisse, Michel Audette, Stéphane Cotin, Stéphane Bordas

To cite this version:

Huu Phuoc Bui, Satyendra Tomar, Hadrien Courtecuisse, Michel Audette, Stéphane Cotin, et al.. Controlling the error on target motion through real-time mesh adaptation: applications to Deep Brain Stimulation. International Journal for Numerical Methods in Biomedical Engineering, John Wiley and Sons, 2017, pp.1-26. 10.1002/cnm.2958 . hal-01672962

HAL Id: hal-01672962
https://hal.archives-ouvertes.fr/hal-01672962

Submitted on 27 Dec 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Controlling the error on target motion through real-time mesh adaptation: applications to Deep Brain Stimulation

Huu Phuoc Bui* [1], Satyendra Tomar [1], Hadrien Courtecuisse [2], Michel Audette [3], Stéphane Cotin [4] and Stéphane P.A. Bordas* [1,5,6]

[1] *Institute of Computational Engineering, University of Luxembourg, Faculty of Sciences Communication and Technology, Luxembourg*
[2] *University of Strasbourg, CNRS, ICube, Strasbourg, France*
[3] *Department of Modeling, Simulation and Visualization Engineering, Old Dominion University, Norfolk, USA*
[4] *Inria Nancy-Grand Est, Villers-lès-Nancy, France*
[5] *Institute of Mechanics and Advanced Materials, School of Engineering, Cardiff University, UK*
[6] *Intelligent Systems for Medicine Laboratory, University of Western Australia, Perth, Australia*

SUMMARY

An error-controlled mesh refinement procedure for needle insertion simulations is presented. As an example, the procedure is applied for simulations of electrode implantation for Deep Brain Stimulation. We take into account the brain shift phenomena occurring when a craniotomy is performed. We observe that the error in the computation of the displacement and stress fields is localised around the needle tip and the needle shaft during needle insertion simulation. By suitably and adaptively refining the mesh in this region, our approach enables to control, and thus to reduce, the error whilst maintaining a coarser mesh in other parts of the domain. Through academic and practical examples we demonstrate that our adaptive approach, as compared to a uniform coarse mesh, increases the accuracy of the displacement and stress fields around the needle shaft, while for a given accuracy, saves computational time with respect to a uniform finer mesh. This facilitates real-time simulations. The proposed methodology has direct implications in increasing the accuracy, and controlling the computational expense of the simulation of percutaneous procedures such as biopsy, brachytherapy, regional anesthesia, or cryotherapy. Moreover, the proposed approach can be helpful in the development of robotic surgeries because the simulation taking place in the control loop of a robot needs to be accurate, and to occur in real time.
Copyright © 2017 John Wiley & Sons, Ltd.

Received 26 September 2017; Revised 7 December 2017; Accepted 27 December 2017

KEY WORDS: real-time simulation; a posteriori error estimate; finite element method; adaptive refinement; deep brain stimulation; brain shift

*Correspondence to: huu-phuoc.bui@alumni.unistra.fr (H. P. Bui), stephane.bordas@alum.northwestern.edu (S. P. A. Bordas)

1. INTRODUCTION

In modern clinical interventions such as biopsy, brachytherapy, cryotherapy, regional anesthesia, or drug delivery, needle-based percutaneous procedures play an important role, thanks to their minimally invasive character. Good training and careful planning to optimise the path to the target, while avoiding critical internal structures, directly influence the desired achievement of these procedures [1]. In some instances, robotic devices can be used to assist these procedures [2]. Unfortunately, natural tissue motion before or during the intervention, and organ deformation due to interactions with the needle during insertion, generally lead to inefficient or incorrect planning [1]. An accurate simulation of needle insertion can address these issues, possibly in a manner complemented by imaging in intraoperative scenarios. For most of the biomechanical problems, in addition to the simulation accuracy, the computational speed is vital since the simulation is a kernel part of an optimised algorithm for the needle trajectory or for a robotic control loop.

In a similar context, Deep Brain Stimulation (DBS) is a surgical procedure which can treat a number of disabling neurological symptoms, e.g. Parkinson's disease (PD). DBS consists of inserting a long needle-shaped electrode (or electrode lead) through a small opening in the skull into the brain to reach the target area. For example, the target area for PD is the subthalamic nucleus (STN). Subsequently, a neurostimulator sends therapeutic electrical impulses to this target, which is implanted with electrodes. The success of the therapy largely depends on the accurate implantation of the electrode lead [3], although DBS targeting increasingly benefits from knowledge of afferent and efferent white matter tracts [4]. This procedure necessitates a pre-operative planning step for determining the target coordinates to implant the electrodes, as well as a trajectory to deliver the insertion cannula, through which then one determines the electrode insertion path. A pre-operative magnetic resonance imaging (MRI) of the patient is employed to identify and locate the target within the brain. However, during the DBS procedure, after the craniotomy is performed, brain shift occurs due to the leak of cerebro-spinal fluid [5, 6]. This includes an error in the planning, which can at best make the therapy ineffective, and at worst cause complications, such as psychiatric disorder [7].

Real-time simulations are increasingly frequent for many applications, from physically based animation [8, 9] to medical simulation [10]. Within the medical context, simulations involving interactions between an interventional radiologist or a surgeon with soft, deformable organs, are conducive to accelerated training and dynamic treatment planning. Indeed, these simulations do not only have the potential to help surgeons train or plan complex operations, but they can also guide them during the intervention itself. In spite of the obvious importance of quality control in such safety-critical advanced numerical tools, no approach is known to evaluate and control the error that occurs during simulations. If this were possible, not only would the user be provided with useful information regarding the quality of the results, but the computational resources could be focused more on large error areas and less on small error areas, particularly where clinically relevant, thus leading to computational savings.

Yet, whilst the literature in error estimation is rich within the computational mechanics community, see e.g., [11–14], little or no work has been done in the context of medical simulation, in particular for real-time applications [15]. The error in numerical simulation comes in three guises. First, users write a mathematical model of the system of interest. For example, they may choose a corotational model to represent the tissue, or a hyper-elastic model with a number of parameters. The inability of the model to represent the physical reality is measured by the *model error*. Second, since analytical solutions are rarely available, the mathematical model must be solved numerically. For example, a user might wish to use the finite element method (FEM), or meshfree methods, or the boundary element method for the numerical solution. The inability of a numerical method to exactly solve the mathematical model leads to the *discretisation error*. Third, the set of linear equations provided by the discretised model must be solved numerically. The associated error is known as the *numerical error* and incorporates, e.g. round-off errors. In this paper, we only tackle the discretisation error which is, in general, much higher than the numerical error (which is presumed to be small as a result of using double precision). Our assumption is that the mathematical model used (here the corotational formulation) properly represents the behaviour of the organ (in our case, the brain) in question. Though hyper-visco-elastic material may represent the brain matter more closely [16], we employ the corotational model for speed and robustness reasons (as it is based on the simple linear elastic model, see Section 2.2). Yet, the tools that we develop in this paper are directly applicable to any model that can be described in the form of partial differential equations, which is the case of most models of soft tissues.

In needle insertion simulations, one has to deal with three main models: a model for the soft tissue, a model for the flexible needle, and a model describing the needle-tissue interaction. Needle and tissue models can rely on linear or non-linear constitutive laws (see the survey in [17]). A nonlinear viscoelastic Kelvin model is used for the needle in [18]. The model describing the interaction between the needle and the tissue remains an important challenge. It consists of a number of different physical aspects, such as frictional contact on tissue surface, puncturing, cutting, sliding with friction, and Poynting's effect. Dehghan et al. [19] propose an ultrasound-based motion estimation to model needle-tissue interaction. To avoid obstacles while steering the needle during insertion, different kinds of needle models have been proposed, for example, a symmetric tip [20], a flexible bevel tip [21–23], as well as a compliant needle model [24]. In our paper, serially-linked Euler-Bernoulli beam elements based on a corotational model are employed to model the needle behaviour, as in [25]. The tissue is modelled using a corotational model as well. Frictional interactions between the needle and the tissue are taken into account, see Section 2.1.

FEM is the most commonly used numerical method for needle insertion simulations, see e.g. [26–28], and is used for simulations presented in this paper. In order to numerically solve the equilibrium equations of mathematical models, FEM uses the concept of discretisation, in which the volume of an object/organ is decomposed into elements (such as tetrahedra or hexahedra). The arrangement of these elements defines the mesh, and a weak form of the continuous mathematical problem, in this case biomechanical deformation, can be numerically integrated on this mesh thanks to shape (interpolation) functions defined on each element of the mesh. The resulting discrete problem (refer to Section 2.2), with the number of unknowns directly proportional to the number of elements in the mesh, can then be easily solved on a computer. It is noted that, for standard methods, the computational cost (time) is also directly proportional (with cubic power) to the number of unknowns, i.e., the CPU time is of the order of N^3, where N denotes the number of unknowns of the problem [29].

The use of a uniform coarse mesh (involving few elements of the same shape) is undermined by the limitation that discontinuities engendered by cuts, singularities or stress concentrations cannot be accurately captured. On the other hand, it is prohibitively expensive to use a uniform fine mesh (involving a large number of small elements, in comparison with the number of elements of a coarse mesh). Therefore, as will be shown in subsequent sections, in the context of real-time simulations, an economical hybrid mesh should be used to deliver a fast simulation response while still enabling accurate results.

The aim of this paper is to demonstrate that error estimation techniques can be effectively used, even in real-time scenario, to control the discretisation error and to drive local refinement of the discretisation. We focus on the problem of needle insertion because of the particular challenges it poses, due to the localisation of deformations and stresses close to the needle shaft. We model the interactions between the needle and the tissue employing an adaptive meshing technique, which is guided by an a posteriori error estimate. An a posteriori error estimation approach obtains the actual error bounds using the computed solution instead of the asymptotic bounds on the solution of the original mathematical problem.

We use an a posteriori error estimation technique called super-convergent patch recovery (SPR). The SPR technique was proposed by Zienkiewicz and Zhu [30], and the asymptotic convergence of this technique is studied in [31, 32]. By solving the discrete problem on a computer, we obtain the numerical solution (henceforth called the *raw* solution, as in [33]). In the SPR (see Section 2.3 for details), an *improved* solution is recovered from the *raw* solution. The difference between the *raw* solution and the *improved* solution is called the approximate error. The key idea is that, where the approximate error is greater than a certain threshold, the mesh should be locally refined, and where the error is small (these two solutions are close together), the mesh can be kept unchanged or coarsened. Note that the mesh refinement is only introduced as a measure to improve the simulation accuracy of the interactions between the needle and the tissue. Similar to the approach used in [25], we do not require the mesh used for the tissue to conform to the needle path. The efficiency of our method will be studied through a heuristic needle insertion scenario, and also through a more complex simulation of the insertion of an electrode lead which is used in DBS. Brain shift prior to electrode insertion will be taken into account in our simulations.

The remaining of the paper is organised as follows. In Section 2, we describe the problem being solved. The interaction model between needle and tissue is formulated, and it is described how the elements are marked for refinement during simulation by employing an error estimate. It is then followed by the discussion on solving the system equations with constraints. Numerical results are presented in Section 3, which shows the efficiency of the proposed approach through a simple needle insertion scenario, as well as through more complex DBS lead insertion simulation. Finally, conclusions are drawn in Section 4.

2. METHODS

We first outline the mathematical model we choose to represent the brain and its interaction with the cannula/needle/lead. We then describe the approach we follow to solve this problem numerically.

We use corotational elasticity, which implicitly assumes that the deformations of the brain are small. We develop a frictional needle-tissue interaction model. The problem is solved using hexahedral finite elements implemented in the open-source Simulation Open Framework Architecture (SOFA, www.sofa-framework.org). The local h-refinement strategy is based upon simple SPR-based a posteriori error estimation method.

2.1. *Problem statement*

Within the context of needle (or cannula/lead) insertion into soft tissue, both the needle and the tissue are modeled as dynamic deformable objects. Let Ω represent the volume of an object (e.g. the tissue). Undergoing external force $\bar{t}$, which is applied on the boundary part Γ_t on the object boundary Γ, and bearing a prescribed displacement $\bar{\mathbf{u}}$ on its boundary part Γ_u in Γ, see Figure 1, the dynamic equilibrium motion equation of the object is expressed by, see [34, 35],

$$\operatorname{div}\boldsymbol{\sigma} + \mathbf{b} + \boldsymbol{\lambda} = \rho\ddot{\mathbf{u}} \qquad \text{in } \Omega. \tag{1}$$

Here $\boldsymbol{\sigma}$ is the Cauchy stress tensor (accounting for internal forces), $\mathbf{b}$ is the body force vector (e.g. gravity), $\boldsymbol{\lambda}$ is the interaction force with another object (e.g. a needle), ρ denotes the mass density, $\mathbf{u}$ stands for the displacement field of the object, and $\ddot{x}$ is the second partial derivative of x with respect to time. In physical terms, the divergence term ($\operatorname{div}\boldsymbol{\sigma}$) is the sum of the derivatives of the stress tensor components with respect to each axis direction. The compatibility conditions on the boundaries Γ_t and Γ_u read

$$\boldsymbol{\sigma}\cdot\mathbf{n} = \bar{\mathbf{t}} \qquad \text{on } \Gamma_t \tag{2a}$$

$$\mathbf{u} = \bar{\mathbf{u}} \qquad \text{on } \Gamma_u. \tag{2b}$$

Equations (2a) and (2b) are known as Neumann and Dirichlet boundary conditions, respectively.

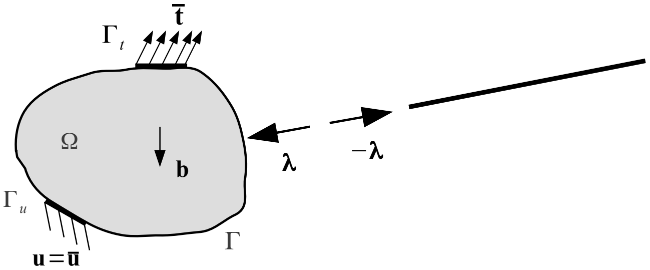


Figure 1. An object Ω undergone a traction force $\bar{\mathbf{t}}$, a body force $\mathbf{b}$, a prescribed displacement $\bar{\mathbf{u}}$, and is in contact with another object through interaction forces $\boldsymbol{\lambda}$.

The strain in the object, which is a measure of the object deformation, can be expressed from the gradient of the displacement as

$$\boldsymbol{\epsilon} = \frac{1}{2}\left(\operatorname{grad}\mathbf{u} + (\operatorname{grad}\mathbf{u})^T\right). \tag{3}$$

The constitutive law, that expresses the relation between stress and strain tensors (through a function f) of the object via the model internal (intrinsic) variables $\boldsymbol{\nu} = (\nu_1, \nu_2, \ldots, \nu_n)$, reads

$$\boldsymbol{\sigma} = f(\boldsymbol{\epsilon}, \boldsymbol{\nu}). \tag{4}$$

In this paper, we employ a linear elastic model based on corotational formulation which has only two mechanical parameters: Young's modulus, and Poisson's ratio. Equations (1) to (4) constitute the governing equations that describe the behaviour of the object in the sense of continuum mechanics.

The interaction force $\boldsymbol{\lambda}$ is defined from the law describing the needle-tissue interaction. During the needle insertion, we prescribe three different types of constraints between the needle and the tissue as in [36], see Figure 2. The Coulomb's friction law is employed to describe the frictional contact within these three types of constraints. First, we define a puncture constraint between the needle tip and the tissue surface. This constraint follows the Kuhn-Tucker conditions. The latter describes the fact that the contact force only exists when the needle tip is in contact with the tissue surface. In the direction n (normal to the tissue surface), this condition reads

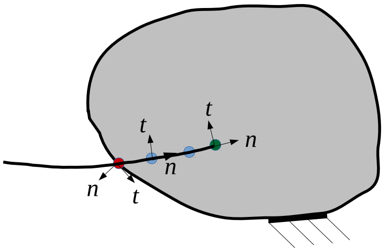


Figure 2. As in [36], three types of constraints between the needle and soft tissue are defined: surface puncture (denoted by $\boldsymbol{\lambda}^{ts}$, illustrated in red), needle tip constraint (denoted by $\boldsymbol{\lambda}^{nt}$, illustrated in green) and needle shaft constraints (denoted by $\boldsymbol{\lambda}^{ns}$, illustrated in blue). To express the constraint directions, we define, at each constraint point, a local coordinate system n-t.

$$\delta_n \geq 0, \qquad \lambda_n^{ts} \geq 0, \qquad \delta_n \cdot \lambda_n^{ts} = 0, \tag{5}$$

where, the superscript ts stands for the constraints on the tissue surface, δ_n is the distance between the needle tip and the tissue surface in the direction n, and λ_n^{ts} is the contact force in the normal (to the tissue surface) direction. Let λ_{p_0} express the puncture strength of the tissue. When the interaction force is greater than the puncture strength of the tissue

$$\lambda_n^{ts} > \lambda_{p_0}, \tag{6}$$

the needle can penetrate into the tissue. And while the needle tip is in contact with the tissue surface but has not penetrated into it yet, the relative motion between them can be described by a frictional law. The Coulomb's friction law is employed to take into account the stick/slip between the needle tip and the tissue surface

$$\lambda_t^{ts} < \mu\lambda_n^{ts} \quad \text{(stick)}; \quad \lambda_t^{ts} = \mu\lambda_n^{ts} \quad \text{(slip)}, \tag{7}$$

in which the subscript t denotes the tangent component of the contact force, μ is the friction coefficient.

Second, as soon as the needle penetrates into the tissue, we define a constraint between the needle tip and the tissue. The fact that the needle tip is stuck or can cut and advance in the tissue depends on the relationship between the normal component (in the direction n, along the needle shaft) and the tangent component (in the direction t) of the contact forces (see Figure 2), defined by

$$\lambda_n^{nt} < \mu\lambda_t^{nt} + \lambda_{c_0} \text{ (stick)}; \quad \lambda_n^{nt} \geq \mu\lambda_t^{nt} + \lambda_{c_0} \text{ (cut and slip)}, \tag{8}$$

where, the superscript nt stands for the constraints at the needle tip, and λ_{c_0} denotes the cutting strength of the tissue.

Finally, we define the needle shaft constraints (denoted by the superscript ns) along the needle shaft. These constraints help to enforce the needle shaft to follow the insertion trajectory created by the advancing needle tip. Again, we employ the Coulomb's friction law to describe the stick and slide contacts between the tissue and the needle shaft, expressed as

$$\lambda_n^{ns} < \mu\lambda_t^{ns} \text{ (stick)}; \quad \lambda_n^{ns} = \mu\lambda_t^{ns} \text{ (slide)}. \tag{9}$$

Note that these three types of constraints describe different kinds of physical interactions between the needle and the tissue. However, we depict three of them as interaction constraint vector, collectively denoted by $\boldsymbol{\lambda}$ in the following, which can be solved numerically.

2.2. *Weak form*

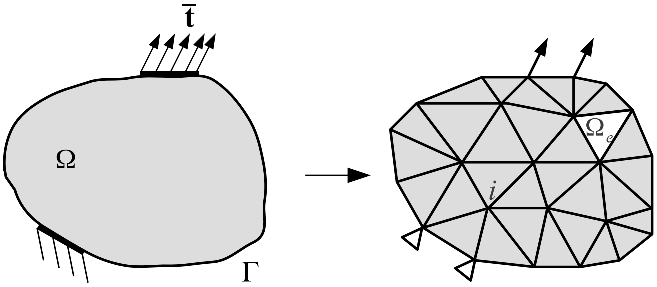


Figure 3. Simplified illustration of FEM discretisation in two dimensions.

Since the partial differential equation (1), which states the equilibrium of the system, involves both spatial and temporal derivatives, it can be solved numerically by discretising that equation in both space (the volume representing the object) and time.

Using N_n nodes, the domain Ω, which represents tissue or needle, is spatially discretised into N_e finite elements Ω_e, $e = 1, 2, \ldots, N_e$, see Figure 3. By integrating the equilibrium equation on each element volume, and assembling for the whole volume, we obtain the discrete problem as (see, e.g., [34, 35])

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f}^{ext} + \mathbf{H}^T\boldsymbol{\lambda}, \tag{10}$$

with $\mathbf{M}$ the mass matrix, $\mathbf{K}$ the stiffness matrix, $\mathbf{C}$ the damping matrix, and $\mathbf{f}^{ext}$ the external force vector. The interaction force constraints $\boldsymbol{\lambda}$, between the needle and the tissue, are computed using Lagrange multipliers (in numerical methods, the constraints $\boldsymbol{\lambda}$ are called Lagrange multipliers, and thus we use the same notation), where $\mathbf{H}^T$ provides the direction of the constraints.

Equation (10) can be rewritten as

$$\mathbf{M}\mathbf{a} = \mathbf{f}(\mathbf{x}, \mathbf{v}) + \mathbf{H}^T\boldsymbol{\lambda}, \tag{11}$$

where $\mathbf{a} = \ddot{\mathbf{u}}$, $\mathbf{v} = \dot{\mathbf{u}}$, $\mathbf{x}$ are the acceleration, velocity and position vectors, respectively, and $\mathbf{f}(\mathbf{x}, \mathbf{v}) = \mathbf{f}^{ext} - \mathbf{K}\mathbf{u} - \mathbf{C}\mathbf{v}$ is the net force (the difference between the external and internal forces) applied to the object. Note that the displacement vector $\mathbf{u}$ is expressed through the current and initial position vectors, $\mathbf{x}$ and $\mathbf{x}_0$, respectively, as $\mathbf{u} = \mathbf{x} - \mathbf{x}_0$.

For temporal discretisation, i.e. to numerically solve the problem in time, we use an implicit backward Euler scheme [37], which is described as follows

$$\ddot{\mathbf{u}}_{t+\tau} = \frac{\dot{\mathbf{u}}_{t+\tau} - \dot{\mathbf{u}}_t}{\tau}; \quad \dot{\mathbf{u}}_{t+\tau} = \frac{\mathbf{u}_{t+\tau} - \mathbf{u}_t}{\tau}, \tag{12}$$

where τ stands for the time step. Inserting Equation (12) into Equation (11) yields the final discrete system

$$\underbrace{\left(\mathbf{M} - \tau\mathbf{C} - \tau^2\mathbf{K}\right)}_{\mathbf{A}} d\mathbf{v} = \underbrace{\tau\mathbf{f}(\mathbf{x}^t, \mathbf{v}^t) + \tau^2\mathbf{K}\mathbf{v}^t}_{\mathbf{b}} + \mathbf{H}^T\boldsymbol{\lambda} \tag{13}$$

or simply $\mathbf{A}d\mathbf{v} = \mathbf{b} + \mathbf{H}^T\boldsymbol{\lambda}$, where $d\mathbf{v} = \mathbf{v}_{t+\tau} - \mathbf{v}_t$. After solving (13) for $d\mathbf{v}$, the position and velocity are updated for needle and tissue as

$$\mathbf{v}_{t+\tau} = d\mathbf{v} + \mathbf{v}_t; \quad \mathbf{x}_{t+\tau} = \mathbf{x}_t + \tau\mathbf{v}_{t+\tau}. \tag{14}$$

In this contribution, a lumped mass matrix, in which a diagonal mass matrix (from the mass density ρ) is integrated over the volume of each element is employed. The stiffness matrix $\mathbf{K}$ is computed based on the corotational FEM, in which the rigid body motion from total finite element displacements is extracted (since it does not contribute to element deformations, see [38]). Note that the corotational FEM formulation makes it possible to handle large rotations for both, needle as well as tissue. For higher accuracy of the computed strain field and less sensitivity to locking phenomena, the soft tissue domain is discretised using hexahedral elements having 3 translational degrees of freedom per node[†]. We use a mesh that does not conform to the boundary of the domain, similar to the approach of Immersed Boundary Method [42] or unfitted finite elements [43, 44], to avoid the complex issue of generating an exact hexahedral mesh of the domain. The needle, on the other hand, is modeled using Euler-Bernoulli beam elements having 6 degrees of freedom (3 translations and 3 rotations) per node.

2.3. Error estimate and adaptive refinement

The accuracy of the outcome of surgical simulators depends on a number of factors. The latter can mainly be divided into two sources: modelling error and discretisation error. The first error source arises when we formulate a mathematical model for a physical problem, prompting the question: does the resulting model correctly describe the physical phenomenon? The second error source comes from the discretisation approach used by numerical methods such as FEM [45], or meshfree methods [46, 47] to solve the mathematical model. In this paper, we assume the mathematical model to be correctly representing reality and focus only on the discretisation error.

By discretising the physical domain in order to solve numerically the mathematical model, the FE method introduces naturally a discretisation error in the solution. We estimate this error source by the superconvergent patch recovery (SPR) procedure [30]. This information is then used to adaptively refine elements where the error is high.

[†] It is also possible to use smoothed tetrahedral elements, which do not lock, see, e.g. [39–41].

The SPR is based on the following simple idea. The displacement field is obtained from solving the equilibrium equation, and the stress field is then computed by differentiating the polynomial approximation of the displacement field. In doing so, one looses numerical accuracy of the stress field (the differentiation reduces the polynomial degree by one). The stress field, obtained by the FEM, is possibly discontinuous across element boundaries. However, assuming that the original problem of the continuum has a smooth (continuous) solution, one can post-process the FEM solution to obtain higher accuracy. This post-processing is done on a collection of neighboring elements, which is called a patch. Using SPR, the post-processed (smoothed) stress field $\boldsymbol{\sigma}^s$ is recovered from the stresses computed at the element centre. The idea of this technique is based on the fact that the stress and strain are more accurate at the superconvergent points (for the case of linear hexahedral elements used in this study, these points are at the centre point of elements), than anywhere else in the element. Using the least squares approach, the values of stress and strain at the superconvergent points are then employed to recover the nodal stress and strain.

The approximate error of an element Ω_e is defined as the energy norm

$$\eta_e = \sqrt{\int_{\Omega^e} (\boldsymbol{\epsilon}^h - \boldsymbol{\epsilon}^s)^T (\boldsymbol{\sigma}^h - \boldsymbol{\sigma}^s) \mathrm{d}\Omega}, \tag{15}$$

in which ϵ and σ are strain and stress tensors, respectively. This error norm is indeed the difference between the FEM solution, denoted by superscript h, and an improved (recovered) solution, denoted by superscript s. An element is marked for refinement if

$$\eta_e \geq \theta \eta_M, \quad 0 < \theta < 1, \text{ where } \eta_M = \max_e \eta_e. \tag{16}$$

Based on a predefined template which is a set of nodes and an associated topology defining template elements, an element being marked for refinement is then replaced by the finer template elements using the mapped mesh method [36, 48]. The template nodes are added by using their natural coordinates (i.e. the coordinates defined on the reference coordinate system with respect to the template). The position $\mathbf{x}_j$, in Cartesian coordinates, of the template node j is computed as

$$\mathbf{x}_j = \sum_{i=1}^{8} \mathbf{x}_i \xi_i^j, \tag{17}$$

where the summation is applied on the eight nodes i of the removed hexahedron element. The term ξ_i^j in the interpolating polynomial (known as the shape function in the FEM context) can be seen as the *relative* position of the node j with respect to the node i. A two dimensional schematic presentation of the refinement can be seen in Figure 4. Note that the predefined template is not limited to regular elements, but can be as heterogeneous as desired (i.e. the template with different element sizes). However, to have a good performance in FEM simulation, ill-shaped elements (i.e. elements where one or more edges are much smaller than the largest) should be avoided in defining the template. For all simulations presented in Section 3, we employ a refinement template $2 \times 2 \times 2$, i.e. each marked element will be refined into eight smaller elements, except where a refinement template is mentioned explicitly. Note also that we are not limited by the regularity of the mesh, and can start from any (reasonable) heterogeneous mesh as a starting point prior to refinement. Also, using this refinement technique, elements can be refined recursively, see Figure 4d.

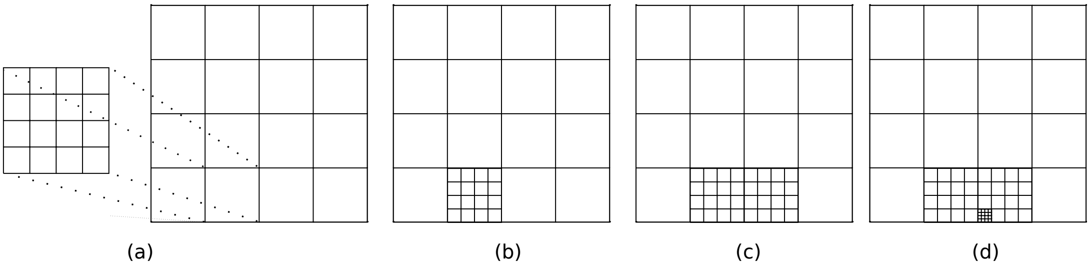


Figure 4. Schematic presentation of template-based refinement: (a) A template for refinement can be defined prior to simulation; (b) first element is refined; (c) second element is refined; (d) the refinement can be nested.

Since elements are refined according to the predefined templates, regardless of their neighbourhood, some nonconforming nodes (also known as hanging nodes) are generated. In FEM words, a nonconforming node is a node which is not shared by all elements around it, see Figure 5. These nodes are also called T-junctions since they form a T-shaped junction in the mesh. Solving the FEM problem without any treatment at T-junctions, displacement discontinuities at these nodes occur. To avoid that, Lagrange multipliers can be used to enforce the continuity at T-junctions, see Section 2.4.

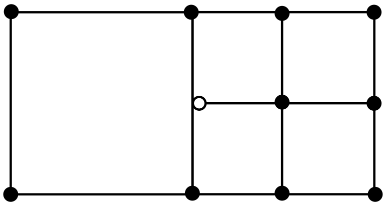


Figure 5. Schematic representation of a hanging node (shown by the unfilled circle) in two dimensions.

2.4. *Solving system equations with constraints*

Based on Equation (13), the interaction between the needle (denoted by subscript 1) and the tissue (denoted by subscript 2) can be expressed by the following equation set

$$\begin{pmatrix} \mathbf{A}_1 & \mathbf{0} & \mathbf{H}_1^T \\ \mathbf{0} & \mathbf{A}_2 & \mathbf{H}_2^T \\ \mathbf{H}_1 & \mathbf{H}_2 & \mathbf{0} \end{pmatrix} \begin{Bmatrix} d\mathbf{v}_1 \\ d\mathbf{v}_2 \\ \boldsymbol{\lambda}_i \end{Bmatrix} = \begin{Bmatrix} \mathbf{b_1} \\ \mathbf{b_2} \\ \mathbf{0} \end{Bmatrix}, \tag{18}$$

where $\boldsymbol{\lambda}_i$ is the Lagrange multiplier representing the *interaction* between the needle and the tissue.

When refinement takes place in the tissue model, T-junctions (as explained above) can be handled by using constraints in the following form

$$\begin{pmatrix} \mathbf{A}_2 & \mathbf{T}^T \\ \mathbf{T} & \mathbf{0} \end{pmatrix} \begin{Bmatrix} d\mathbf{v}_2 \\ \boldsymbol{\lambda}_t \end{Bmatrix} = \begin{Bmatrix} \mathbf{b_2} \\ \mathbf{0} \end{Bmatrix}. \tag{19}$$

where $\boldsymbol{\lambda}_t$ stands for Lagrange multipliers used for *T-junctions*, and $\mathbf{T}$ expresses the dependence between the T-junction nodes (slave nodes) with respect to their parent nodes (master nodes).

We can see that Equations (18) and (19) have the same general form

$$\begin{pmatrix} \mathbf{A} & \mathbf{J}^T \\ \mathbf{J} & \mathbf{0} \end{pmatrix} \begin{Bmatrix} \mathbf{x} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{b} \\ \mathbf{0} \end{Bmatrix}. \tag{20}$$

Equation (20) can be reformulated as

$$\mathbf{x} = \underbrace{\mathbf{A}^{-1}\mathbf{b}}_{\mathbf{x}_{free}} - \underbrace{\mathbf{A}^{-1}\mathbf{J}^T\boldsymbol{\lambda}}_{\mathbf{x}_{cor}}, \tag{21a}$$

$$\underbrace{\mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T}_{\mathbf{W}}\boldsymbol{\lambda} = \mathbf{J}\,\underbrace{\mathbf{A}^{-1}\mathbf{b}}_{\mathbf{x}_{free}}, \tag{21b}$$

in which, $\mathbf{x}_{free}$ can be seen as the solution of the unconstrained system $\mathbf{A}\mathbf{x} = \mathbf{b}$, and $\mathbf{x}_{cor}$ is the corrective solution due to constraints, and can be obtained by solving $\mathbf{A}\mathbf{x}_{cor} = \mathbf{J}^T\boldsymbol{\lambda}$, $\mathbf{W} = \mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T$ is the compliance matrix. To be able to solve Equation (20) in real time, we employ the advanced algorithm described in Algorithm 1.

Algorithm 1 Algorithm for solving the system equation with constraints (Equation (20)). $\mathbf{L}$ denotes a lower unit triangular matrix, and $\mathbf{D}$ is a diagonal matrix. The function $solveCholesky$, described in Algorithm 2, is used to solve efficiently an equation set. The computation of the compliance matrix $\mathbf{W}$ is described in Algorithm 3.

```
1: $\mathbf{LDL}^T \leftarrow decompose(\mathbf{A})$                                   ▷ (Cholesky decomposition)
2: $\mathbf{x}_{free} \leftarrow solveCholesky(\mathbf{A}\mathbf{x}_{free} = \mathbf{b})$          ▷ (Solve unconstrained problem)
3: $\boldsymbol{\lambda} \leftarrow solve(\mathbf{W}\boldsymbol{\lambda} = \mathbf{J}\mathbf{x}_{free})$                 ▷ (Solve for constraints, see Equation (21b))
4: $\mathbf{x}_{cor} \leftarrow solveCholesky(\mathbf{A}\mathbf{x}_{cor} = \mathbf{J}^T\boldsymbol{\lambda})$          ▷ (Compute corrective solution)
5: $\mathbf{x} \leftarrow \mathbf{x}_{free} - \mathbf{x}_{cor}$                          ▷ (Update solution, see Equation (21a))
```

We perform Cholesky decomposition of matrix $\mathbf{A}$ described in Line 1 of Algorithm 1 on GPU using NVIDIA GPU-accelerated libraries. Note that since the size of the matrix $\mathbf{J}$ is much smaller than that of the matrix $\mathbf{A}$, and thus the size of the matrix $\mathbf{W} = \mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T$ is small, a direct solve of $\mathbf{W}\boldsymbol{\lambda} = \mathbf{J}\mathbf{x}_{free}$ described in Line 3 does not pose any significant computational overhead.

Algorithm 2 Given Cholesky decomposition $\mathbf{LDL}^T$ of $\mathbf{Q}$, this algorithm solves the equation $\mathbf{Q}\mathbf{w} = \mathbf{p}$.

```
$\mathbf{y} \leftarrow solve(\mathbf{L}\mathbf{y} = \mathbf{p})$                    ▷ (Forward substitution)
$\mathbf{w} \leftarrow solve(\mathbf{D}\mathbf{L}^T\mathbf{w} = \mathbf{y})$               ▷ (Back substitution)
```

Algorithm 3 Given Cholesky decomposition $\mathbf{LDL}^T$ of $\mathbf{A}$, this algorithm computes the compliance matrix $\mathbf{W} = \mathbf{J}\mathbf{A}^{-1}\mathbf{J}^T$.

```
1: $\mathbf{S} = \mathbf{L}^{-1}\mathbf{J}^T$
2: $\mathbf{T} = \mathbf{D}^{-1}\mathbf{S}$
3: $\mathbf{W} = \mathbf{S}^T\mathbf{T}$
```

As proposed in [10], we parallelise the computations described in Lines 1 and 3 of Algorithm 3 on GPU. The computation in Line 2 of Algorithm 3 is inexpensive since $\mathbf{D}$ is diagonal. It is noted that, to be even faster in simulations, Courtecuisse and coworkers [10] have proposed an asynchronous preconditioner $\mathbf{P} \approx \mathbf{A}$ which can be updated at low frequency on a dedicated CPU thread to solve a linear system (Line 2 of Algorithm 1), and to compute the compliance matrix $\mathbf{W}$ described in Algorithm 3. However, this approach could introduce simulation errors since the system matrix $\mathbf{A}$ is not taken into account properly in the resolution of the system at each time step. Therefore, in this paper, in order to focus on the discretisation error, the system matrix $\mathbf{A}$ is appropriately considered in the solution at each time step, and such an asynchronous preconditioner is not employed.

3. RESULTS

In this section, we present several numerical studies in order to demonstrate the efficiency of our method. To show the benefits of employing adaptive refinement controlled by error estimates, we first study the simulation of needle insertion into a phantom tissue of simple geometry. Then, we study a more complex simulation of deep-brain stimulation (DBS) lead insertion. In DBS, a cannula is inserted into the brain tissue in order to reach an STN target. Leaving an electrode inside the brain, the cannula is then pulled back until completely outside of the brain. By these studies, we provide insight into the mechanical behaviour of the brain tissue in response to the lead insertion, and analyse the effect of mesh adaptivity on the solution.

DBS is an effective approach to alleviate the symptoms of neuro-degenerative diseases such as Parkinson's disease. During most DBS interventions, the craniotomy leads to a shift of the brain within the skull, because of the cerebrospinal fluid which is drained out of the skull cavity and ceases to provide buoyancy to the brain. Once the electrode has been inserted, the brain regains its original position.

One of the difficulties during DBS interventions is that the electrode displaces as the brain comes back to its original position, before brain shift. Therefore, it is possible that the electrode tip displaces, so that the electrode does not stimulate the proper target anymore, which requires a new operation.

Simulations of the type that we are presenting here have the potential to help surgeons prepare for interventions by investigating where the electrode tip should be located, so that it stimulates the target *after the brain shifts back to the configuration it had prior to craniotomy-induced brain shift*.

3.1. Impact of local mesh refinement on displacement field

The effect of adaptive refinement on the compromise between the computational time and the solution accuracy is studied through a needle insertion simulation into a phantom tissue. In this study, the mechanical properties are taken as follows. The Young's modulus is set to 50 MPa for the needle, and 10 MPa for the tissue, whereas 0.4 and 0.3 are set for the Poisson's ratio of the tissue and the needle, respectively. It is noted that, for this heuristic study, these parameters are chosen arbitrarily (but are physically meaningful for a needle-tissue interaction problem). Figure 6 shows an illustration of the needle insertion problem.

As mentioned above, we employ a linear elastic model based on corotational formulation to model the behaviour of the needle as well as the tissue. The tissue geometry has the dimension of $4 \times 2 \times 2$ cm. The length of the needle and its radius are of 3.2 cm and 0.1 cm, respectively.

The mesh is subsequently refined in a uniform manner ($8 \times 4 \times 4$, $16 \times 8 \times 8$, $32 \times 16 \times 16$ elements), as well as in an adaptive manner. By the adaptive refinement scheme, we start simulations

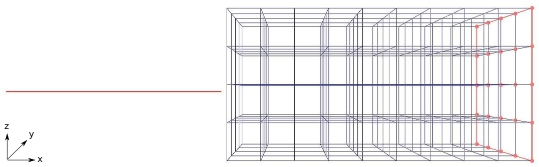


Figure 6. Schematic illustration of the needle insertion problem into the tissue with a simple geometry being clamped at the right end surface as the boundary condition.

with the coarse mesh ($8 \times 4 \times 4$ nodes) and adaptively refine the mesh during the simulation thanks to the error estimate, see Section 2. During needle insertion, we measure the displacement of the points defined in Figure 7.

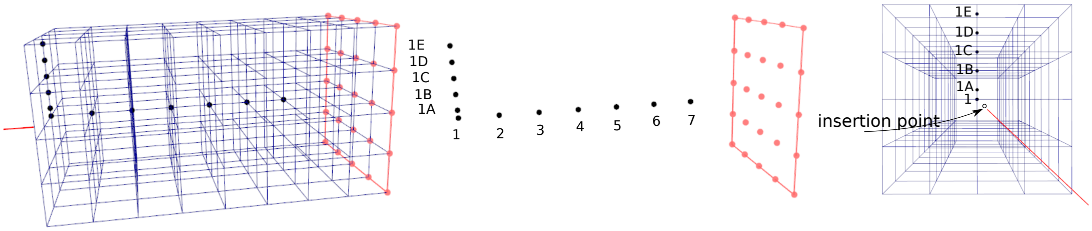
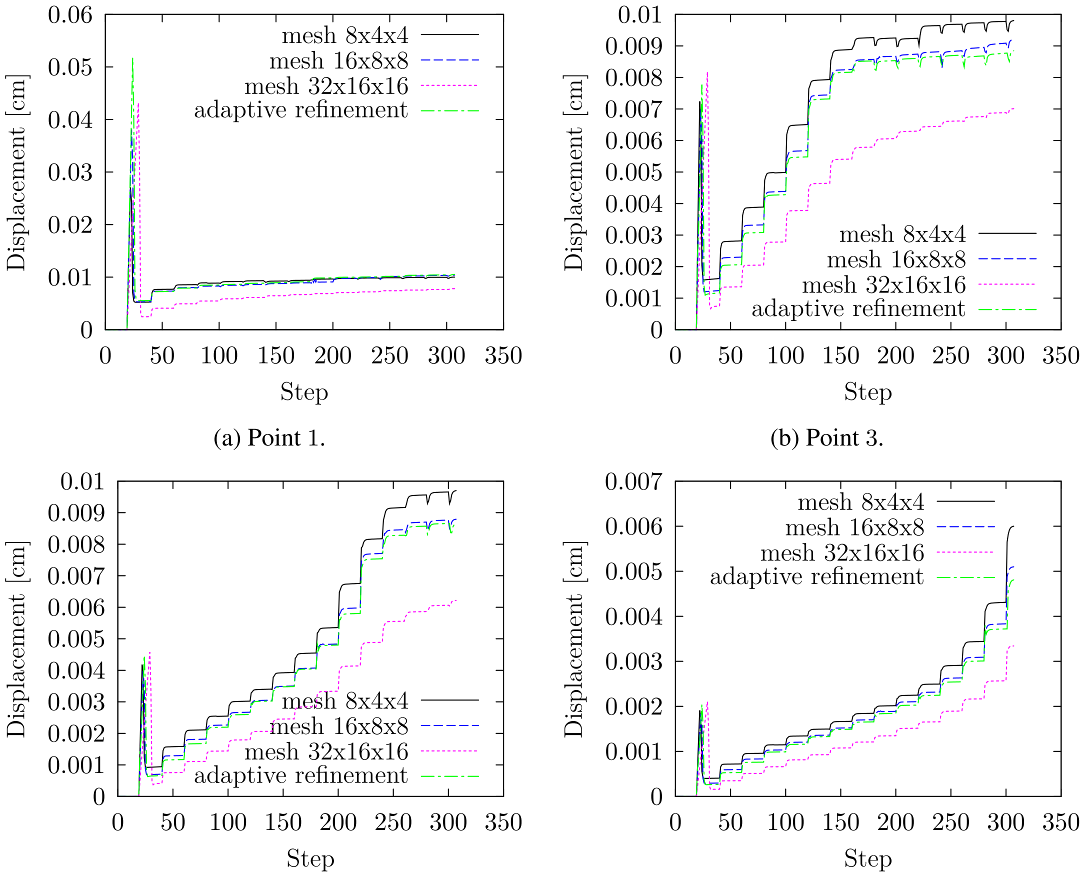


Figure 7. The displacement during insertion is measured at the predefined points, namely, $1, 2, 3, 4, 5, 6, 7$ which are distributed in the direction of the needle shaft, and $1A$, $1B$, $1C$, $1D$, $1E$ that lie in the direction normal to the needle shaft.

Note that the displacements of all points is not shown here. Figure 8 shows the displacement of some points distributed in the direction of the needle shaft, and Figure 9 shows the displacement of some points distributed in the direction normal to the needle shaft (i.e. radial). These results reveal two distinct features. Firstly, the displacement solution of the adaptive refinement scheme agrees well with that of fine mesh when the adaptivity criterion is appropriately chosen (here $\theta = 0.3$). Secondly, it is observed that after puncture, the finer the mesh is used, the smaller the displacements at those points are. This observation is explained by two phenomena. First, a finer mesh produces softer behaviour for the tissue, i.e. the displacement does not “propagate” as easily from the needle shaft to the point where we measure the displacement (refer to [36] for details). Second, needle insertion leads to localisation of displacements, strains and stresses, due to the small size of the needle compared to the organ.

Figure 10 shows the displacement at all predefined points during insertion when mesh $32 \times 16 \times 16$ is employed. It is seen that the further the point from the tissue surface is (Figure 10a), or the further the point from the needle shaft is (Figure 10b), the smaller the displacement of that point is. The physical relevance of this observation is explained in the previous paragraph.

Remark In our study, we use three names to differentiate between uniform meshes. We call the uniform mesh $8 \times 4 \times 4$ as the *coarse mesh*, the uniform mesh $16 \times 8 \times 8$ as the *fine mesh*, and the uniform mesh $32 \times 16 \times 16$ as the *very fine mesh*. The results of the adaptive refinement procedure shown in Figures 8 and 9 agree well with those of the fine mesh, but differ from those of the very fine mesh. This is simply due to the fact that we use a mesh template $2 \times 2 \times 2$ for refinement (i.e. one element is refined to eight sub-elements) for the simulation. Since the uniform mesh $16 \times 8 \times 8$ is two times finer than the one $8 \times 4 \times 4$, using the adaptive refinement procedure



(a) Point 1.

(b) Point 3.

(c) Point 5.

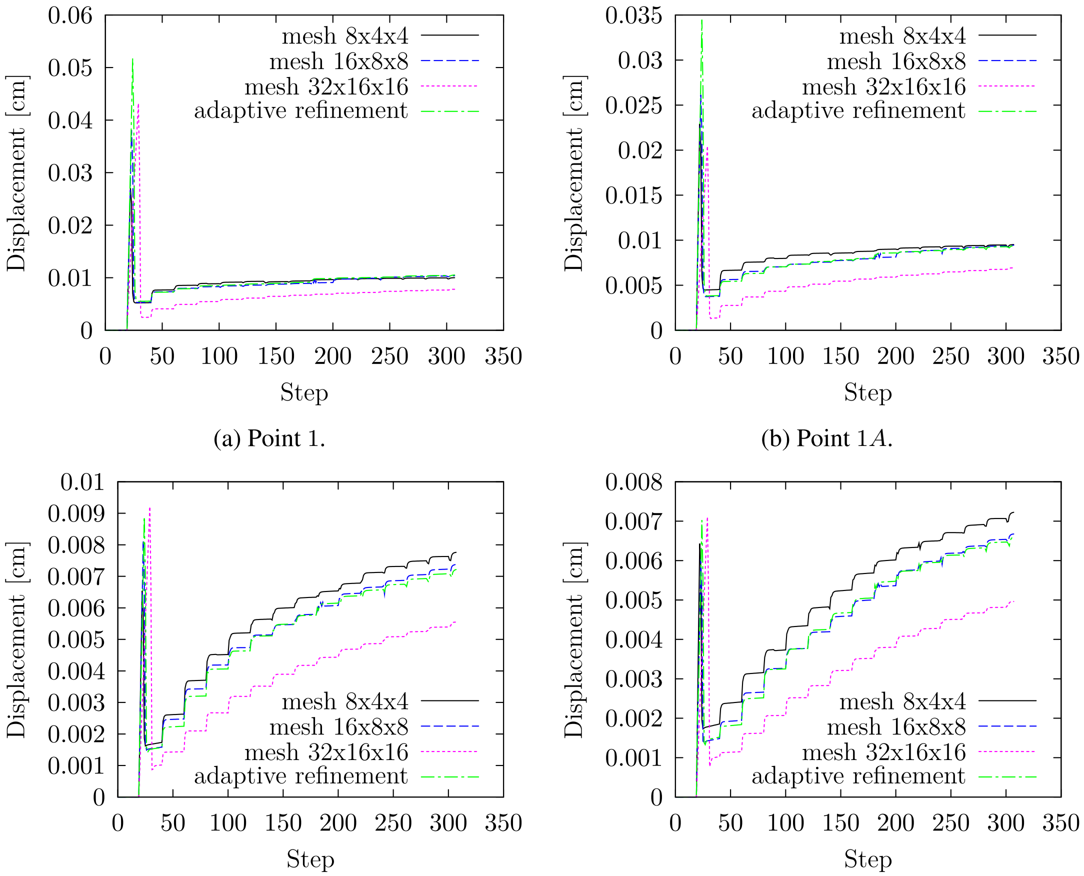
(d) Point 7.

Figure 8. During needle insertion, the displacement of the points located in the direction of the needle shaft, uniform and adaptive refinement schemes. The numbers of degrees of freedom for the meshes listed in the graphs (in the label order) are 675, 4 131, 28 611, and 1 461 (for adaptive refinement scheme it is the maximum at the end of insertion step). It is seen that the displacement solution of adaptive refinement scheme agrees well with that of fine mesh when certain criterion of adaptivity is appropriately chosen (here $\theta = 0.3$). Moreover, it is observed that the finer the mesh is used, the smaller the displacements at those points are.

with the refinement template $2 \times 2 \times 2$ will provide the solutions which agree with those of the mesh $16 \times 8 \times 8$. As shown in Figure 11, when using a refinement template $4 \times 4 \times 4$, the results of the adaptive refinement scheme agree well with those of the very fine mesh ($32 \times 16 \times 16$). And if we use an intermediate template $3 \times 3 \times 3$ for refinement, the solutions will be in between the solution of coarse and very fine meshes (see Figure 11).

3.2. Electrode implantation in DBS

We study the simulation of DBS lead placement by inserting a cannula into a brain model until it reaches the predefined STN target. Then, the cannula is retracted while keeping an electrode inside the brain. As in [5], the Young's modulus of 10 GPa is set for the cannula and for the electrode, while the Young's modulus of the brain tissue is set to 6 kPa. The Poisson's ratio is set as 0.45 for the brain tissue, and 0.3 for both the cannula and the electrode. The radii of the cross sections are set to 3 mm and 0.7 mm for the cannula and electrode, respectively. The frictional coefficient between the



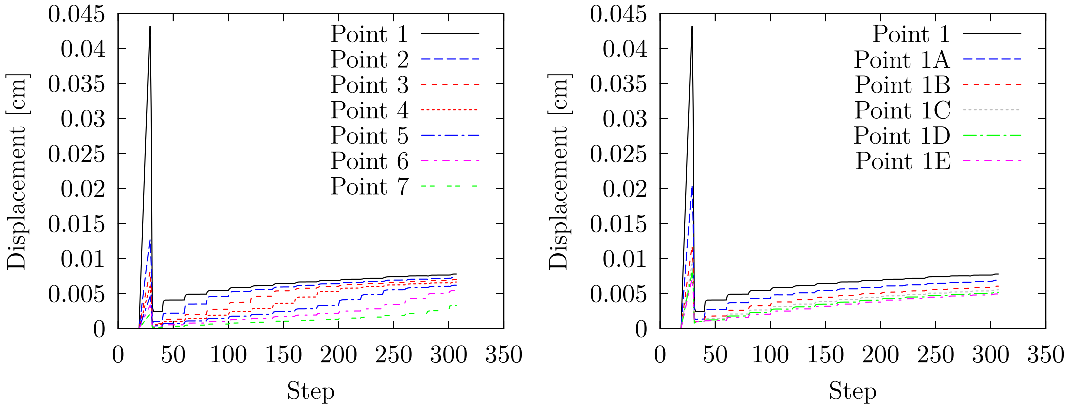
(a) Point 1.

(b) Point $1A$.

(c) Point $1C$.

(d) Point $1E$.

Figure 9. The displacement of the points located in the radial direction, uniform and adaptive refinement schemes, during needle insertion. The numbers of degrees of freedom for the meshes listed in the graphs (in the label order) are 675, 4 131, 28 611, and 1 461 (for adaptive refinement scheme it is the maximum at the end of insertion step). The same observations can be made as in Figure 8.



(a) In needle direction.

(b) In radial direction.

Figure 10. Displacement of the points during needle insertion, when the 32 × 16 × 16 mesh is employed. The further the point from the tissue surface, or from the needle shaft, is found, the smaller the displacement of that point is.

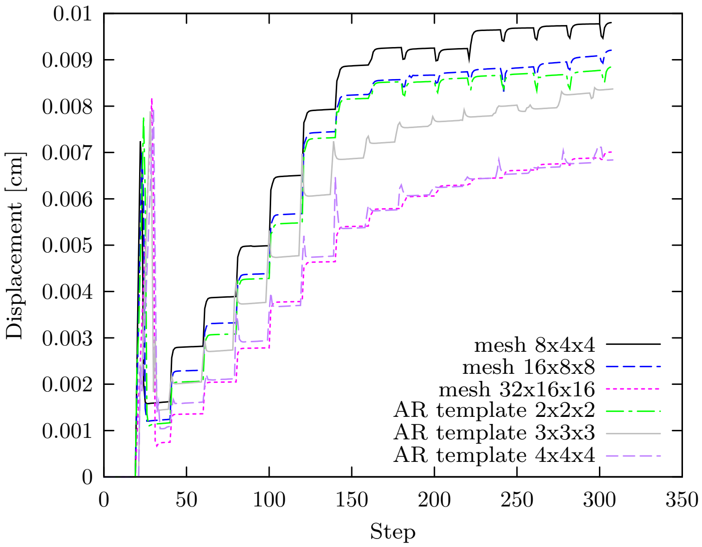


Figure 11. Displacement measured at the Point 3 during needle insertion simulation. AR stands for "Adaptive refinement". It is shown that when a finer template mesh ($4 \times 4 \times 4$) is employed for the adaptive refinement scheme, the results provided by the adaptive refinement procedure agree well with those of the uniform finest mesh ($32 \times 16 \times 16$).

cannula and the brain tissue is established at 0.05. The adaptivity parameter θ is set to 0.6. From our studies, we found that, with minimum DOFs, this choice of the adaptivity parameter gives the best results for DBS simulation of electrode placement. Since the brain tissue is very soft, we set the penetration strength at the brain surface and the cutting strength to 0.01 N.

It is worth mentioning that, in this study, simple boundary conditions around the brain tissue are taken into account. Indeed, we simply consider clamped bilateral constraints around the area of the optic nerves and the brainstem. We believe that this choice of boundary conditions (for the sake of simplicity) does not affect the efficiency of the adaptive refinement algorithm. However, more complex boundary conditions for the brain with respect to the skull can be found in [5, 49]. We are currently investigating the effects of the choice of boundary conditions on the results for this particular case. In general, for brain shift problems, it is now known that the type of boundary conditions does not have a strong effect on the resulted brain shift, see, e.g., [50, 51]. However, systematic studies are still missing from the literature for the problem of DBS simulation.

Two strategies of mesh refinement are employed. The uniform refinement consists of refining each hexahedron of the mesh into 8 smaller hexahedra. In the second strategy, called adaptive refinement, we only refine the elements that fulfil the adaptivity condition Equation (16).

In Figure 12, brain shift prior to insertion is shown. Figure 13 shows the distance between the electrode tip and the STN target during insertion and retraction of the cannula. The vertical line in Figure 13 indicates the point in time when the cannula has the trend to be pulled back. It is observed that from that time, the distance between the electrode tip and the STN target still decreases. This is justified by the fact that the brain tissue is unloaded from insertion forces and displaces toward the direction from which the needle was inserted. The electrode, on the other hand keeps moving forward due to inertial forces. When the cannula is being retracted, the brain tissue deforms

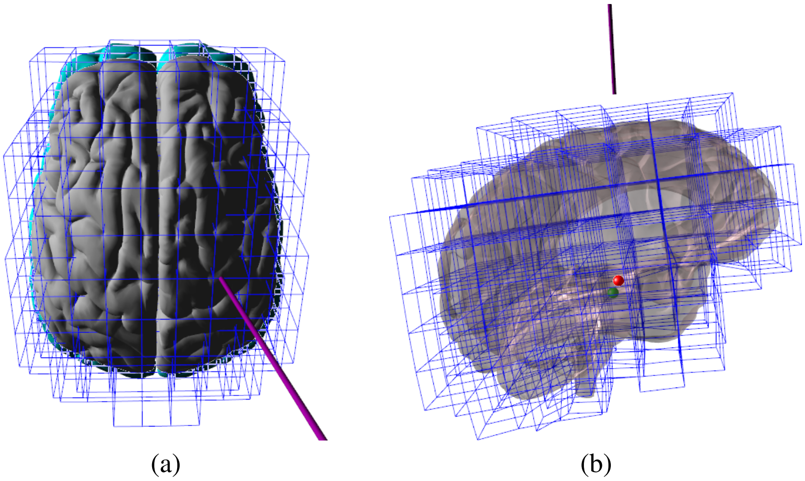


Figure 12. Simulation of brain shift prior to insertion of the cannula for electrode implantation. Initial position of the brain is shown by cyan colour (a). The position of the STN target after brain shift is shown by the green sphere, while its initial position is shown by the red one (b). The displacement of the STN target due to brain shift is about 6.5 mm.

and moves backward, and thus increases the electrode-STN target distance accordingly. When the cannula is completely outside the brain tissue, that distance is stabilised at around 2.9 mm.

Moreover, Figure 13 reveals that the distance from the electrode tip to the STN target obtained by using our adaptive refinement scheme agrees well with that when a fine uniform mesh is employed. However, an important gain in computational time can be observed, since the maximum number of DOFs for the adaptive scenario is only $3\,135$ as compared to $12\,528$ DOFs of the fine uniform mesh. This reduction in the problem size is approximately of the order of 4.

Figure 14 shows the displacement of the STN target (with respect to its position after brain shift) when inserting the cannula, leaving the electrode, and retracting the cannula. It can be seen that at around the step 75, the cannula starts penetrating into the brain tissue, and thus provokes the displacement of the STN target inside the brain. This displacement increases along with the insertion of the cannula and the electrode into the brain, due to cutting of the brain tissue and friction on the cannula shaft. At the step 166, when the cannula starts undergoing retraction, the brain tissue is then unloaded, and this causes the STN target to move backward. This phenomenon is seen by the decrease of the STN target displacement up to the step between 200 and around 225 (depending on the mesh employed, refer to Figure 14). After that, since the cannula is bearing retraction, and because of friction, the brain tissue follows the cannula in the direction of the retraction. As a consequence, the displacement of the STN target increases again. The closer to the brain surface the cannula tip is, the smaller the magnitude of frictional force along the cannula shaft is, and therefore, at some stage, the STN target displacement decreases again. And, that displacement continuously decreases when the cannula is fully retracted from the brain tissue, as observed at the final stage of Figure 14.

Visualisation of the simulation of DBS lead implantation during insertion and retraction at some stages can be seen in Figure 15. Figure 14 reveals that, during the insertion phase, the result of the adaptive mesh does not closely agree with that of the uniform fine mesh. This is evidenced by the fact that the refinement has begun but has not reached the STN target yet, as can be partly seen in Figure 15b. However, from the stage when the cannula reaches the STN target, to the end of

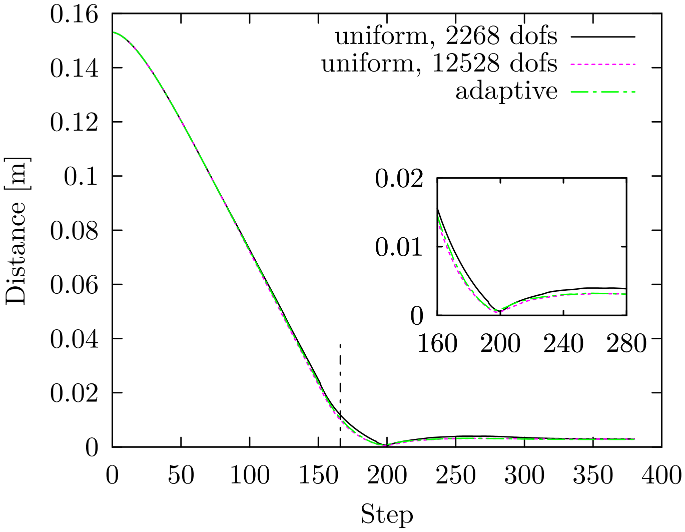


Figure 13. The distance from the electrode tip to the STN target. An STN target is fixed inside the brain. The cannula is inserted into the brain so that it can reach the STN target. Then the electrode is left while retracting the cannula. The vertical line at step 166 indicates the moment when the cannula has the trend to be pulled back. It is seen that using an adaptive refinement scheme, the distance obtained between the electrode tip and the STN target is as good as using the fine uniform mesh. However, a significant gain in computational time is observed. The maximum number of degrees of freedom for the adaptive refinement is 3,135. For a similar solution profile, compared to a uniformly refined mesh, this represents a reduction in the problem size of approximately 4.

the simulation when it is completely pulled out of the brain tissue, the result of the adaptive mesh agrees well with that of the uniform fine mesh, thanks to the refinement occurring along the cannula trajectory, as seen in Figure 15d.

It is again observed that the use of the proposed adaptive refinement scheme is very relevant because it guarantees the solution accuracy close to the fine meshing schemes, with computational time suitable for real-time simulation.

4. CONCLUSIONS

We have presented a structured study to answer the essential, but rarely addressed, question of accuracy in surgical simulations. The original contribution of our paper is the use of an a posteriori estimate of the discretisation error to automatically drive local adaptive mesh refinement during needle insertion.

From the study on the displacement solution of two different problems (needle insertion into a simple phantom geometry with stiff mechanical properties, and the simulation of electrode implantation for deep brain stimulation (DBS) with softer mechanical properties for the brain), two major conclusions can be drawn:

- The outcome of the proposed adaptive refinement scheme does not depend on the mechanical parameters of the tissue, such as Young's modulus and Poisson's ratio.
- With a suitably chosen refinement criterion, the adaptive solution agrees well with the solution on a fine uniform mesh, while saving computational time to make it feasible for real-time

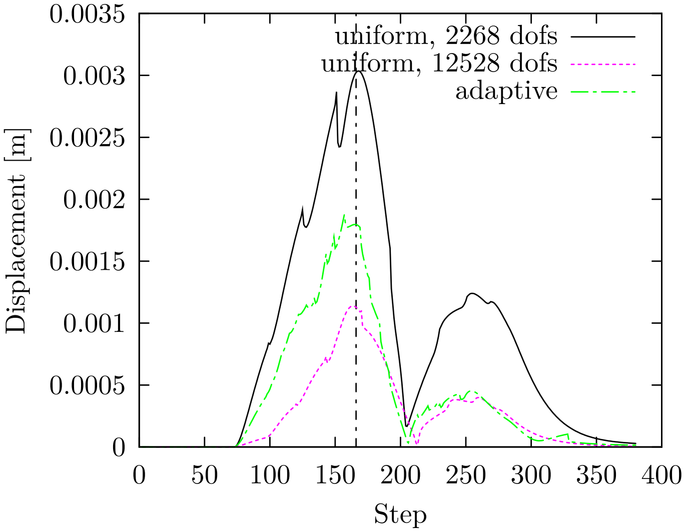


Figure 14. Displacement of STN target (with respect to its position after brain shift) during cannula insertion and retraction (the same simulations as in Figure 13). The vertical line, same as in Figure 13, shows the stage when the cannula is going to be removed. The maximum number of DOFs for the adaptive refinement is 3135.

simulations. The computational time savings in the particular case we tackled are of the order of 10 (more details can be found in [36]).

Our work is limited in several ways which we are currently investigating:

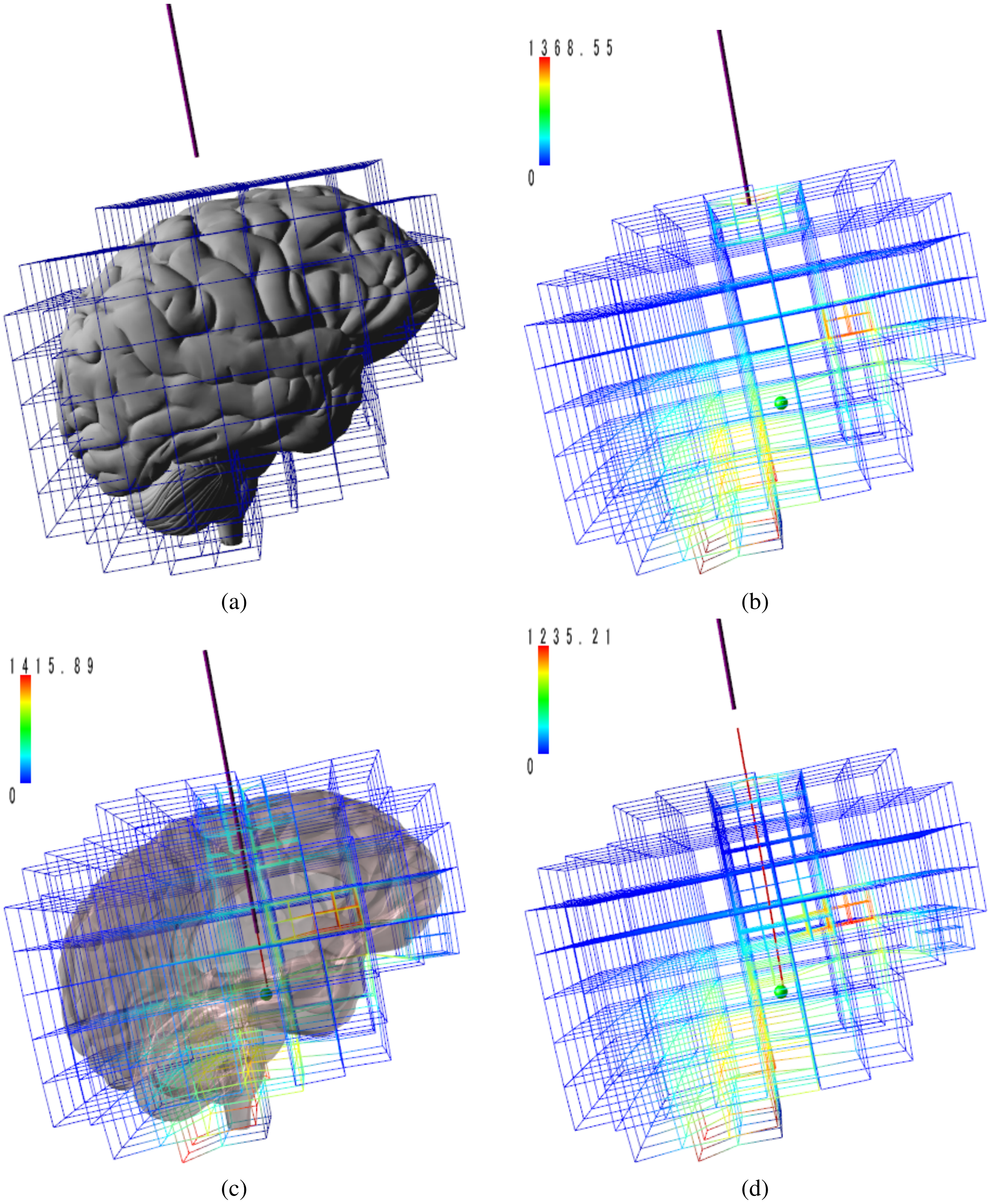
- We tackled only the discretisation error, and avoided the difficult problem of "model error". For example, we assumed the brain to behave in a corotational manner. Although this is corroborated by the literature [5,52–54], we do not have physical evidence that this assumption is true in general, nor as to when it may break down.

- We assumed simple boundary conditions around the brain. This should be investigated in more detail to assess the effect of both geometrical and boundary condition uncertainty on the outcome of the simulations.

- Soft tissue properties vary from patient to patient by up to a few orders of magnitude. We have been working on robust and systematic approaches to quantifying these uncertainties [55]. We will use these methods to quantify the effects of variability in the material parameters on the motion of the target and of the electrode during DBS.

5. SUPPLEMENTARY MATERIAL

A movie file of needle insertion simulations, in which the error is estimated during the simulations and drives automatically mesh adaptation, is submitted together with this article.



(a) (b) (c) (d)

Figure 15. Visualisation of the electrode implantation simulation of DBS (a). Brain shift occurs prior to cannula insertion. The STN target during the simulation is shown at different stages by a green sphere in (b,c,d). When the cannula tip is in contact with the brain tissue surface, it leads to element refinement guided by error estimate, shown in (b), coloured by Von Mises stress magnitude. When the cannula has reached the STN target, it undergoes a retraction. The cannula is being retracted while the electrode is left inside the brain, shown in (c). The cannula is completely retracted, adaptive refinement has occurred along the trajectory of the cannula, shown in (d).

ACKNOWLEDGMENTS

Stéphane Bordas, Satyendra Tomar and Huu Phuoc Bui thank partial funding for their time provided by the European Research Council Starting Independent Research Grant (ERC Stg grant agreement No. 279578) RealTCut “Towards real time multiscale simulation of cutting in non-linear materials with applications to surgical simulation and computer guided surgery". We are also grateful for the funding from the Luxembourg National Research Fund (INTER/MOBILITY/14/8813215/CBM/Bordas and INTER/FWO/15/10318764).

Stéphane Cotin thanks for funding of European project RASimAs (FP7 ICT-2013.5.2 No. 610425).

The funding from the University of Strasbourg Institute for Advanced Study (BPC 14/Arc 10138) for the first author is gratefully acknowledged.

Huu Phuoc Bui thanks Dr. Davide Baroli and Mr. Vahid Shahrabi for their useful helps about PyMesh (https://github.com/qnzhou/PyMesh) and Meshmixer (http://www.meshmixer.com).

The authors would also like to thank Dr. Alexandre Bilger and Mr. Rémi Bessard Duparc for their general helps.

REFERENCES

1. Hamzé N, Peterlík I, Cotin S, Essert C. Preoperative trajectory planning for percutaneous procedures in deformable environments. *Computerized Medical Imaging and Graphics* 2016; **47**:16–28.
2. Elgezua I, Kobayashi Y, Fujie MG. Survey on current state-of-the-art in needle insertion robots: Open challenges for application in real surgery. *Procedia CIRP*, vol. 5, 2013, doi:10.1016/j.procir.2013.01.019.
3. Breit S, Schulz JB, Benabid AL. Deep brain stimulation. *Cell and Tissue Research* 2004; **318**(1):275–288, doi: 10.1007/s00441-004-0936-0.
4. Calabrese E. Diffusion Tractography in Deep Brain Stimulation Surgery: A Review. *Frontiers in Neuroanatomy* 2016; **10**:45, doi:10.3389/fnana.2016.00045.
5. Bilger A, Dequidt J, Duriez C, Cotin S. *Biomechanical Simulation of Electrode Migration for Deep Brain Stimulation*. Springer Berlin Heidelberg: Berlin, Heidelberg, 2011; 339–346, doi:10.1007/978-3-642-23623-5_43.
6. Hamz N, Bilger A, Duriez C, Cotin S, Essert C. Anticipation of brain shift in deep brain stimulation automatic planning. *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015; 3635–3638, doi:10.1109/EMBC.2015.7319180.
7. Piasecki SD, Jefferson JW. Psychiatric complications of deep brain stimulation for parkinson's disease. *The Journal of clinical psychiatry* 2004; **65**(6):845–849.
8. Nealen A, Müller M, Keiser R, Boxerman E, Carlson M. Physically based deformable models in computer graphics. *Computer graphics forum*, vol. 25, 2006; 809–836.
9. Wang Y, Jacobson A, Barbič J, Kavan L. Linear subspace design for real-time shape deformation. *ACM Transactions on Graphics (TOG)* 2015; **34**(4):57.
10. Courtecuisse H, Allard J, Kerfriden P, Bordas SP, Cotin S, Duriez C. Real-time simulation of contact and cutting of heterogeneous soft-tissues. *Medical Image Analysis* 2014; **18**(2):394 – 410.
11. Rabczuk T, Belytschko T. Adaptivity for structured meshfree particle methods in 2d and 3d. *International Journal for Numerical Methods in Engineering* 2005; **63**(11):1559–1582, doi:10.1002/nme.1326.
12. Nguyen-Thanh N, Muthu J, Zhuang X, Rabczuk T. An adaptive three-dimensional rht-splines formulation in linear elasto-statics and elasto-dynamics. *Computational Mechanics* Feb 2014; **53**(2):369–385, doi:10.1007/s00466-013-0914-z.
13. Nguyen-Thanh N, Zhou K, Zhuang X, Areias P, Nguyen-Xuan H, Bazilevs Y, Rabczuk T. Isogeometric analysis of large-deformation thin shells using rht-splines for multiple-patch coupling. *Computer Methods in Applied Mechanics and Engineering* 2017; **316**(Supplement C):1157 – 1178, doi:https://doi.org/10.1016/j.cma.2016.12.002. Special Issue on Isogeometric Analysis: Progress and Challenges.
14. Anitescu C, Hossain MN, Rabczuk T. Recovery-based error estimation and adaptivity using high-order splines over hierarchical t-meshes. *Computer Methods in Applied Mechanics and Engineering* 2018; **328**(Supplement C):638 – 662, doi:https://doi.org/10.1016/j.cma.2017.08.032.
15. Wu X, Downes MS, Goktekin T, Tendick F. Adaptive nonlinear finite elements for deformable body simulation using dynamic progressive meshes. *Computer Graphics Forum* 2001; **20**(3):349–358, doi:10.1111/1467-8659.00527.
16. Mihai LA, Chin L, Janmey PA, Goriely A. A comparison of hyperelastic constitutive models applicable to brain and fat tissues. *Journal of The Royal Society Interface* 2015; **12**(110), doi:10.1098/rsif.2015.0486.

17. Abolhassani N, Patel R, Moallem M. Needle insertion into soft tissue: A survey. *Medical Engineering and Physics* 2007; **29**:413–431.
18. Mahvash M, Dupont PE. Fast Needle Insertion to Minimize Tissue Deformation and Damage. *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, ICRA'09, IEEE Press: Piscataway, NJ, USA, 2009; 2761–2766.
19. Dehghan E, Wen X, Zahiri-Azar R, Marchal M, Salcudean S. Needle-tissue interaction modeling using ultrasound-based motion estimation: Phantom study. *Computer Aided Surgery* 2008; **13**(5):265–280.
20. Heverly M, Dupont P, Triedman JK. Trajectory Optimization for Dynamic Needle Insertion. *ICRA*, IEEE, 2005; 1646–1651.
21. Alterovitz R, Goldberg K, Okamura A. Planning for steerable bevel-tip needle insertion through 2d soft tissue with obstacles. *Proceedings of ICRA 2005*, 2005; 1652–1657.
22. Webster R, Kim J, Cowan N, Chirikjian G, Okamura A. Nonholonomic modeling of needle steering. *International Journal of Robotics Research* 2006; **5**(6):509–525.
23. Misra S, Reed K, Schafer B, Ramesh K, Okamura A. Mechanics of flexible needles robotically steered through soft tissue. *Int. J. Rob. Res.* Nov 2010; **29**(13):1640–1660.
24. Tang L, chen Y, He X. Compliant needle modeling and steerable insertion simulation. *Computer-Aided Desing and Applications* 2008; **5**:39–46.
25. Duriez C, Guébert C, Marchal M, Cotin S, Grisoni L. Interactive simulation of flexible needle insertions based on constraint models. *Lecture Notes in Computer Science*, vol. 5762, 2009; 291–299.
26. DiMaio S, Salcudean S. Needle insertion modeling and simulation. *IEEE Transactions on Robotic Automation* 2003; **19**(5):864–875.
27. Hing JT, Brooks AD, Desai JP. Reality-based needle insertion simulation for haptic feedback in prostate brachytherapy. *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, 2006; 619–624, doi:10.1109/ROBOT.2006.1641779.
28. Jiang S, Hata N, Kikinis R. Needle insertion simulation for image-guided brachytherapy of prostate cancer. *2008 2nd International Conference on Bioinformatics and Biomedical Engineering*, 2008; 1682–1685, doi: 10.1109/ICBBE.2008.749.
29. Watkins DS. *Fundamentals of Matrix Computations*. John Wiley & Sons, Inc., 2005, doi:10.1002/0471249718.ch1.
30. Zienkiewicz O, Zhu J. The superconvergent patch recovery (SPR) and adaptive finite element refinement. *Computer Methods in Applied Mechanics and Engineering* 1992; **101**(1):207–224.
31. Carstensen C, Bartels S. Each averaging technique yields reliable a posteriori error control in FEM on unstructured grids. I. Low order conforming, nonconforming, and mixed FEM. *Math. Comp.* 2002; **71**(239):945–969 (electronic), doi:10.1090/S0025-5718-02-01402-3.
32. Bartels S, Carstensen C. Each averaging technique yields reliable a posteriori error control in FEM on unstructured grids. II. Higher order FEM. *Math. Comp.* 2002; **71**(239):971–994 (electronic), doi:10.1090/S0025-5718-02-01412-6.
33. Bordas S, Duflot M, Le P. A simple error estimator for extended finite elements. *Communications in Numerical Methods in Engineering* 2008; **24**(11):961–971, doi:10.1002/cnm.1001.
34. Zienkiewicz O, Taylor R. *The Finite Element Method: Solid mechanics*. Referex collection.Mecánica y materiales, Butterworth-Heinemann, 2000.
35. Liu GR, Quek SS. Chapter 3 - Fundamentals for Finite Element Method. *The Finite Element Method (Second Edition)*, Liu GR, , Quek SS (eds.). Second edition edn., Butterworth-Heinemann: Oxford, 2014; 43–79, doi: http://dx.doi.org/10.1016/B978-0-08-098356-1.00003-5.
36. Bui HP, Tomar S, Courtecuisse H, Cotin S, Bordas S. Real-time error control for surgical simulation. *IEEE Transactions on Biomedical Engineering* 2017; **PP**(99):1–1, doi:10.1109/TBME.2017.2695587.
37. Baraff D, Witkin A. Large steps in cloth simulation. *Proceedings of SIGGRAPH*, 1998; 43–54.
38. Felippa C, Haugen B. A unified formulation of small-strain corotational finite elements: I. theory. *Computer Methods in Applied Mechanics and Engineering* 2005; **194**(2124):2285 – 2335.
39. Nguyen-Xuan H, Rabczuk T, Nguyen-Thanh N, Nguyen-Thoi T, Bordas S. A node-based smoothed finite element method with stabilized discrete shear gap technique for analysis of reissner–mindlin plates. *Computational Mechanics* 2010; **46**(5):679–701, doi:10.1007/s00466-010-0509-x.
40. Lee CK, Mihai LA, Hale JS, Kerfriden P, Bordas SP. Strain smoothing for compressible and nearly-incompressible finite elasticity. *Computers & Structures* 2017; **182**:540 – 555, doi:http://doi.org/10.1016/j.compstruc.2016.05.004.
41. Mendizabal A, Bessard Duparc R, Bui HP, Paulus CJ, Peterlik I, Cotin S. Face-based smoothed finite element method for real-time simulation of soft tissue 2017, doi:10.1117/12.2255064.
42. Pinelli A, Naqavi IZ, Piomelli U, Favier J. Immersed boundary method for generalised finite volume and finite difference navier-stokes solvers. *Journal of Computational Physics* 2010; **229**(24):9073–9091.

43. Belytschko T, Parimi C, Mos N, Sukumar N, Usui S. Structured extended finite element methods for solids defined by implicit surfaces. *International Journal for Numerical Methods in Engineering* 2003; **56**(4):609–635, doi:10.1002/nme.686.
44. Burman E, Claus S, Hansbo P, Larson MG, Massing A. Cutfem: Discretizing geometry and partial differential equations. *International Journal for Numerical Methods in Engineering* 2015; **104**(7):472–501, doi:10.1002/nme.4823.
45. Zienkiewicz O, Taylor R, Zhu J. *The finite element method: Its basis and fundamentals*, vol. 1. Elsevier, 2013.
46. Nguyen VP, Rabczuk T, Bordas S, Duflot M. Meshless methods: a review and computer implementation aspects. *Mathematics and computers in simulation* 2008; **79**(3):763–813.
47. Griebel M, Schweitzer MA. *Meshfree Methods for Partial Differential Equations*. Springer-Verlag Berlin Heidelberg, 2003.
48. Grosland NM, Bafna R, Magnotta VA. Automated hexahedral meshing of anatomic structures using deformable registration. *Computer Methods in Biomechanics and Biomedical Engineering* 2009; **12**(1):35–43, doi:10.1080/10255840802136143. PMID: 18688764.
49. Wittek A, Miller K, Kikinis R, Warfield S. Patient-specific model of brain deformation: Application to medical image registration. *Journal of biomechanics* 2007; .
50. Wittek A, Kikinis R, Warfield SK, Miller K. *Brain Shift Computation Using a Fully Nonlinear Biomechanical Model*. Springer Berlin Heidelberg: Berlin, Heidelberg, 2005; 583–590, doi:10.1007/11566489_72.
51. Joldes GR, Wittek A, Miller K. Computation of intra-operative brain shift using dynamic relaxation. *Computer Methods in Applied Mechanics and Engineering* 2009; **198**(4144):3313 – 3320, doi:http://doi.org/10.1016/j.cma.2009.06.012.
52. Dequidt J, Coevoet E, Thins L, Duriez C. Vascular Neurosurgery Simulation with Bimanual Haptic Feedback. *Workshop on Virtual Reality Interaction and Physical Simulation*, Jaillet F, Zara F, Zachmann G (eds.), The Eurographics Association, 2015, doi:10.2312/vriphys.20151337.
53. Sase K, Fukuhara A, Tsujita T, Konno A. Gpu-accelerated surgery simulation for opening a brain fissure. *ROBOMECH Journal* Dec 2015; **2**(1):17, doi:10.1186/s40648-015-0040-0.
54. Morin F, Courtecuisse H, Reinertsen I, Lann FL, Palombi O, Payan Y, Chabanas M. Brain-shift compensation using intraoperative ultrasound and constraint-based biomechanical simulation. *Medical Image Analysis* 2017; **40**(Supplement C):133 – 153, doi:https://doi.org/10.1016/j.media.2017.06.003.
55. Hauseux P, Hale JS, Bordas SP. Accelerating monte carlo estimation with derivatives of high-level finite element models. *Computer Methods in Applied Mechanics and Engineering* 2017; :–doi:http://dx.doi.org/10.1016/j.cma.2017.01.041.