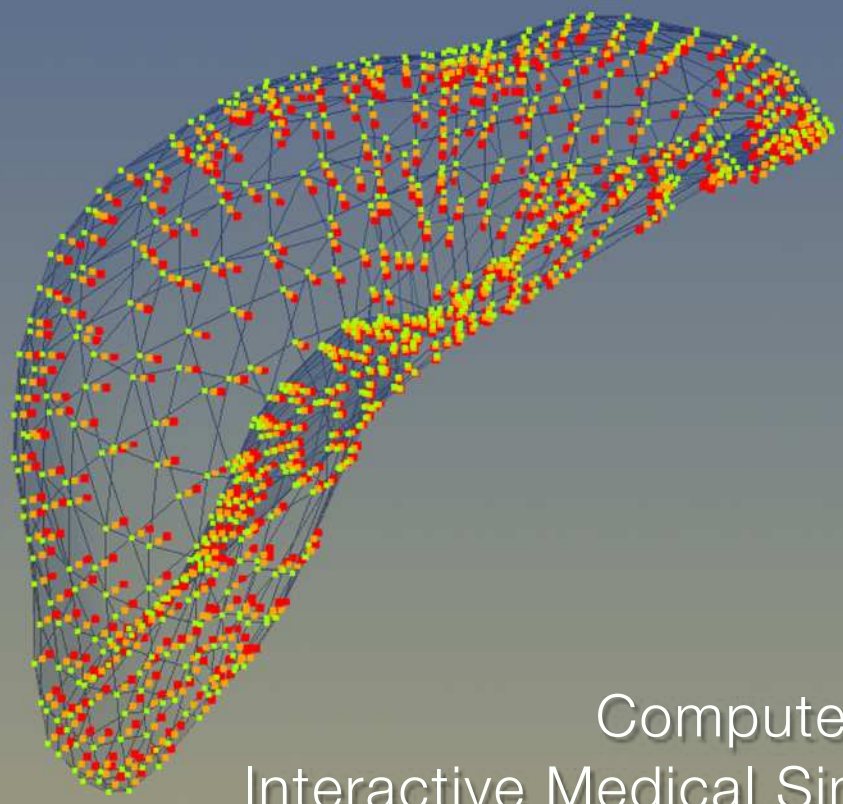
Computer Based Interactive Medical Simulation

Stéphane Cotin

HABILITATION A DIRIGER DES RECHERCHES



Computer Based Interactive Medical Simulation

Stéphane Cotin

Ecole doctorale de Mathématiques

Université de Lille

Jury

Rapporteurs	Dimitris Metaxas, Ph.D. Claude Puech, Ph.D. Gabor Szekely, Ph.D.
Examinateurs	Nicholas Ayache, Ph.D. Steven Dawson, M.D. Sophie Tison, Ph.D.
Directeur	Christophe Chaillou, Ph.D.

Chapter I

Chapter II — Models

Chapter III — Interaction

Chapter IV — Validation

Foreword

The work presented in this manuscript covers about ten years of research, which was almost entirely done in the area of computer-based medical simulation. When I started, there were only a handful of scientists and physicians believing that one day clinicians would be able to improve their skills or learn new techniques using a computer system. Ten years later, many things have changed. The field has grown and many researchers are now working on various aspects of this vast problem. But also several commercial training systems have been developed and are now being used all around the world. Of course, a lot of work remains to be done, and teaching in Medicine is still largely based on the same principles that were introduced decades ago. Yet, new means of training or rehearsing will undoubtfully continue to be developed to improve patient care and change Medicine to offer safer, personalized care.

This manuscript is divided in two parts. The first one highlights a series of scientific contributions, shows directions for future work, and introduces ideas for unifying some of the key elements of interactive simulations. The second part of the manuscript provides a summary of my scientific and supervision activities. Given the particular path I took after my PhD, it is important to remember that those ten years were not entirely dedicated to the type of scientific activities that are usually covered in such a document.

Needless to say, most of the work presented here was not done alone, but through collaborations with incredibly talented people. I am particularly indebted to all the present and past members of the Sim Group at CIMIT, with whom I worked for seven years, for allowing me to work in such a fruitful and diverse environment. Through their ideas, comments and criticisms they have helped turn new ideas into contributions of scientific value and often into actual prototypes of simulators. Among them, I should particularly mention the postdoctoral fellows with whom I closely worked: Vincent Luboz, Christian Duriez, Julien Lenoir, Jeremie Allard, Jeremie Dequidt, and Pierre-Frederic Villard. I also want to thank the physicians who contributed to this work: Nicholas Stylopoulos for his insight into laparoscopic surgery, James Rabinov for assessing on a regular basis our progress in interventional radiology simulation, and David Rattner, head of the department of Surgery at the Massachusetts General Hospital, for helping us with the development of the CELTS training system. But this work would not have been possible with the constant support and the friendship of a wonderful team: Xunlei Wu, Paul Neumann, Ryan Bardsley, Mark Ottensmeyer, Joseph Samoski, Ferol Vernon, Bob Waddington and Rhonda Young, and a wonderful leader: Steve Dawson.

I would also like to express my gratitude to the reviewers of this manuscript: Dimitris Metaxas, Claude Puech, and Gabor Szekely. I would also like to thank the other members of the jury, Nicholas Ayache, Christophe Chaillou, Steve Dawson and Jacques Marescaux.

Last but not least, I dedicate this manuscript to Agnes who has been my best supporter through all these years, and has always been by my side during these times of doubt that every researcher faces once in a while.

Summary

The principal motivation for my research has been that computer-based medical simulation offers significant advantages in increasing the effectiveness of training healthcare personnel while also reducing risk exposure to patients. Computer-based training systems can be effectively deployed across a wide range of expertise, from unexperienced students to advanced practitioners, and trainee performance can be objectively quantified. But simulation for training is only one aspect of medical simulation. Planning and rehearsal of complex interventions, personalized medicine based on patient-specific pre-operative data, and augmented reality in the operating room, are just other aspects of what computer-based simulations can contribute.

To achieve high-fidelity simulation (i.e. to provide the highest possible level of visual, biomechanical, and haptic realism) many problems need to be solved. While the overall challenge lies in the trade-off between interactivity and accuracy of the simulation, it is the multidisciplinary aspect of this field that is the source of most difficulties. Through the different sections of chapters II, III, and IV I will describe a series of contributions in various areas of medical simulation. In particular I will address the importance of anatomical modeling as a mean to create realistic, and potentially patient-specific, virtual representations of the operating field. Through several examples I will emphasize the difficulty of creating accurate computer models of the anatomy that are compatible with the requirements of real-time simulation. I will also address the challenge of determining soft tissue characteristics, and defining mathematical models of soft tissue deformation. In this context also, the constraint of real-time computation requires to simplify, adapt, optimize the models to allow interactivity of the simulation. It becomes therefore essential to design validation methodologies to assess the optimized model and quantitatively compare various approaches against each other. I will also present a series of contributions related to the modeling of tissue-tool interactions. Whether rigid instruments are used to manipulate soft tissues or flexible devices navigate inside anatomical structures, this interaction is an essential part of a medical procedure. Modeling such interactions, through efficient collision detection, and advanced contact mechanics, represents also an important challenge. I will also present elements of research that are less studied in the field of medical simulation, but that play a key role in the overall acceptation of a simulator by the medical community. Among these factors are the visual realism through advanced rendering techniques, the modeling of complex medical devices, the validation of biomechanical models, and the integration of performance metrics to assess the efficiency of the simulator, from an educational standpoint.

In chapter (V) I will describe several prototypes of simulators I have co-developed. Although time-consuming, building these prototypes was an enlightening experience. Not only it provided a mean to validate the various algorithms of a simulation in a unique environment, but it also became a support to collect feedback from the medical community, and ultimately to change the way physicians perceive our work and its potential.

Chapter (VI) will be dedicated to a description of our most recent efforts to design a framework for unifying key elements of real-time interactive simulations. This framework, called SOFA, is a multi-institutional collaborative work which has the potential to accelerate research, development and collaborations in the field of medical simulation.

Finally, chapter (VII) will summarize our different contributions to the field of medical simulation and present challenging and exciting perspectives, in particular in the area of patient-specific simulation.

Chapter I
Interactive Medical Simulation

1. Introduction

The first medical simulators were simple models of human patients. Hundreds of years ago, these representations in clay and stone were used to demonstrate clinical features of disease states and their effects on the anatomy. Similar models are still used today to help students learn for instance the anatomy of the musculoskeletal system. Active models, which attempt to reproduce living anatomy or physiology, are recent developments, with one of the first known system designed around 1950. These mannequins, or patient simulators, are able to recreate several of the key characteristics of human physiology and can respond to physical examination or electrocardiography for instance. Such systems are essentially integrated into training centers that aim at recreating the operating room environment. Initially used for anesthesiology training, these simulators have not technologically evolved over the past decade besides improvements related to their robustness or portability. To compensate for some of their limitations, both in terms of cost effectiveness and versatility, computer-based training systems have been introduced recently. Such systems offer, theoretically, an elegant solution to this problem since they can provide a realistic and configurable training environment, thus bridging the gap between basic training and performing the actual intervention on patients, without any restriction for repetitive training. Increasing computational power, as well as current achievements in the field of interactive computer graphics and virtual reality, have already led to the rapid development of reasonably sophisticated simulation systems during the past ten years.

In this manuscript, I will focus essentially at computer-based interactive medical simulation and will try to paint a clear picture of its related challenges, our key contributions in this area, and possible evolutions envisioned for this field. Although the field of computer-based medical simulation emerged only a decade ago, it has been a source of growing interest both in the scientific and clinical community. Computer-based simulations have the advantage of allowing a student to make judgments, and also to make errors. The process of iterative learning through assessment, evaluation, decision making, and error correction creates a much stronger learning environment than passive instruction. This is particularly relevant in the context of a landmark report published in 2000 by the Institute of Medicine in the United States. Entitled "To Err is Human" (Kohn *et al.*, 2000), this report placed medical errors as the third leading cause of death in the United States, and one of its recommendation was the following: "*Recommendation 8.1: Health care organizations should make continually improved patient safety a declared and serious aim by establishing patient safety programs with defined executive responsibility; and establish interdisciplinary team training programs for providers that incorporate proven methods of training, such as simulation.*" More recently, a few important decisions have contributed to bring Medical Simulation in the limelight. In 2005, the Food and Drug Administration required the use of simulation technology before employing a new medical device for carotid stenting. The creation of the Advanced Initiatives for Medical Simulation (AIMS) conference, the involvement of the Cardiovascular and Interventional Radiological Society of Europe (CIRCE) in the introduction of simulation systems in the early phases of the medical curriculum, and the development of several commercial simulators, are also important elements for the development

of medical simulation in the world. It is therefore essential that we are able to respond, from a scientific standpoint, to this increasing interest. For this to happen, simulators need to become more than a three-dimensional interactive textbook, they need to be a part of the practice of medicine. Simulators also need to address the constantly evolving requirements of modern medicine and the increasing expectations from patients. At a time of growing promises of personalized medicine, the idea of planning and then virtually rehearsing a procedure before operating on the patient becomes a promising area of application for simulation techniques.

2. Related work

It is virtually impossible to list all the work done in areas relevant to computer-based medical simulation, since this field covers so many aspects. The objective of this section is essentially to illustrate, through a variety of examples, the complexity of some of the problems, the variety of the approaches taken, and the difficulty of integrating the various solutions into a prototype or product. These challenges have hindered the development of both prototypes at an academic level, and products at an industrial level. This section could have been organized in many different way: chronologically, by models, or by application; I have chosen to categorize it based on clinical areas in medicine that have been the focus of simulation work.

Laparoscopic surgery

Since its early stages, interactive medical simulation has been essentially targeted towards training. In particular, computer-based medical simulation has known an important development with the widespread introduction of laparoscopic surgery in Medicine. The rationale for this increased interest comes from the need to train surgeons (novice or experienced) to this new technique. At the same time, the higher level of technology that characterizes laparoscopy over surgery, makes it more suitable for being simulated. In particular, the indirect vision of the operating field, using a miniaturized camera, and the indirect manipulation of the tissues, using long instruments transmitting reduced haptic feedback, makes it possible to propose plausible simulation techniques. The simulator in this case consists usually of a haptic device dedicated to laparoscopic surgery, and a monitor showing a virtual representation of the operative field. This makes it theoretically possible to reproduce accurately any laparoscopic procedure. In practice, many challenges remain to be solved: real-time[1] simulation of the deformation of anatomical structures, simulation of cutting, dissection, suturing and other tissue manipulations that modify the topology of the deformable model, collision detection and contact response to describe tissue-tool interactions, or haptic feedback. All of these different components of the simulation are constrained by the need for real-time computation to enable interactivity of the simulation. In addition, creating anatomical models, such that they support the various requirements of collision detection, rendering, or biomechanical modeling, is also an interesting problem for which no real solution exists.

[1] The term “real-time” is often used as a synonym of “interactive” to describe computational models that can respond immediately to the actions of the user. Strictly speaking, real-time means that one second of simulation corresponds to one second of real life. Interactivity is a less restrictive notion and most methods discussed in this manuscript, although often called “real-time”, are only interactive.

The most successful application (even at a commercial level) of simulation techniques for laparoscopy is in the area of surgical skills training. The advantage of skills training over procedural training is that it does not require realistic models of the anatomy or even of the deformable behavior of anatomical structures. In the case of the MIST-VR system (Seymour *et al.*, 2002) (Sickle *et al.*, 2005), simple geometric shapes have been used, as the end goal of such simulators is to teach laparoscopic skills in no particular anatomical context. However, most skills trainers tend to recreate a typical operative field with a few deformable structures. The objective of the simulation is often to reach certain targets using laparoscopic instruments, to perform basic tissue manipulations (Kühnapfel *et al.*, 2000) including suturing (Wang *et al.*, 2005) (SimSurgery).

When full laparoscopic surgery simulation is the objective, the realism of soft-tissue manipulation is a priority. Consequently, a majority of researchers has focused on the computation, in real-time, of the deformation of various organs. Occasionally, simulations of soft tissue dissection has been attempted. The need for interactive simulations has led to a rather large variety of models of deformation, all of which can be called “physics-based” models, but that often lack solid foundations in continuum mechanics or biomechanics. The following list of previous work is not exhaustive, its main purpose is merely to illustrate the variety of representations used to describe soft-tissue deformation.

In early works, a great interest has been given to spring-mass models due to their simplicity of implementation and their relatively low computation complexity. For instance, Jambon *et al.* (Jambon *et al.*, 1998) used such a representation for a gynecology simulation which objective is to teach spatial localization and instrument manipulation within the abdominal cavity. A series of exercises are proposed with increasing degree of difficulty and the simulator can be customized to simulate different scenarios. Raghupathi *et al.* (Raghupathi *et al.*, 2004) developed a prototype of training system for colon cancer removal. The method for animating the small intestine and the mesentery is based on a spring-mass model using damped springs. A stochastic approach for fast collision detection, well-suited for highly deformable self-colliding objects, is also used in this system. The proposed approach, although based on a *ad-hoc* deformable model, led to visually plausible simulations.

In (Basdogan *et al.*, 1998), a Free Form Deformation representation was used to model the behavior of the liver in their laparoscopic simulator. The efficiency of this representation allows them to integrate a force feedback device in the simulator. Suzuki *et al.* (Suzuki *et al.*, 2001) defined a particle system for their liver surgery simulator, coupled with a three-finger force feedback device. Although it allows to perform surgical maneuvers such as pushing, pinching and incising, the proposed representation is not based on any known biomechanical model.

In (Cotin *et al.*, 1999), we proposed a prototype of laparoscopic simulator for hepatic surgery based on a very efficient implementation of a finite element approach to solve linear elastic deformations of a detailed model of the liver. This approach allows realistic deformations as long as they remain small. Also, very fast computation of contact forces is possible, which permits the addition of a dedicated laparoscopic haptic device in the prototype. This initial prototype was completed by the work of Picinbono *et al.* (Picinbono *et al.*, 2001) who proposed an implementation of a finite element method for a non-linear elastic behavior. Basic topological changes, representing the action of a electrical cautery device, were implemented.

In (De *et al.*, 2001), the authors proposed a prototype of a laparoscopic training system where the deformable organs are modeled using a meshless technique. Meshless finite element methods are a

rather new approach which does not require creating an explicit mesh of the volume of the organ. Instead, only a cloud of points, distributed within the volume of the object, is necessary. Shape functions are then built at each node using a neighborhood information. When topological changes need to be handled, such a representation offers a theoretical benefit compared to conventional finite element approaches. It is no longer required to update a mesh when cutting or dissection is simulated. The proposed approach, however, is less computationally efficient than other finite element techniques, thus limiting the simulation to coarse representations of the anatomy.

A final example comes from the work done at ETH Zurich on the development of a surgical training simulator for hysteroscopy (Bachofen *et al.*, 2006) (Tuchschmid *et al.*, 2007). The objective of this project is to go beyond rehearsal of basic manipulative skills, and enable training of procedural skills, involving decision making and problem solving. Through this project, various research areas have been addressed and reported in scientific articles. They include areas such as soft tissue deformation, collision detection and response, and cutting, but also blood flow simulation or haptic feedback.

Arthroscopic surgery

This type of procedure has similarities with laparoscopic surgery, however during arthroscopy, the joint is visualized and accessed through small portals. An optical endoscope equipped with a video camera allows visualization of the procedure through one of the portals, while surgical probes and other instruments are inserted into additional portals The ability to perform diagnosis and surgical intervention in joints without open surgery has had a large impact in orthopedics. Worldwide, several millions of arthroscopic procedures are performed every year. Arthroscopic procedures have been shown to reduce costs, and increase patient recovery rates. However, arthroscopy suffers from specific technical limitations: namely limited visibility through the arthroscope, difficulty in orienting the camera with the surgeon's viewpoint, and restricted motion of the surgical tools. Because of these technical challenges, it is important that surgeons receive adequate training in arthroscopic techniques. Such training could be enhanced by computer simulation. The simulator of knee arthroscopic surgery developed by Gibson *et al.* (Gibson *et al.*, 1998) takes into account the volumetric nature of the organs with a deformation law derived from spring-mass models. The model, called ChainMail, relies on a regular grid of voxels linked to their nearest neighbors. When one node of the structure is pulled or pushed, neighboring links absorb the movement by taking up slack in the structure. If a link between two nodes is stretched or compressed to its limit, displacements are transferred to neighboring links. In this way, small displacements of a selected point in a relatively slack system result in only local deformations of the system, while displacements in a system that is already stretched or compressed to its limit causes the whole system to move. Much like the links in a chain, neighbors only respond to a given nodes movement if the constraints on distances between nodes are violated. Tissue elasticity is modeled in a second process that adjusts the distance between local neighboring volume elements to minimize a local energy constraint. During interactions such as tissue manipulation, only the first part of the deformation algorithm takes place, elastic behavior is only enabled once interactions are reduced or stopped. As with several other techniques described above, such methods allow interactive rates, but lack realism during the deformation process, mostly because anatomical structures are represented as a discrete set of masses rather than a volume.

Open surgery

The difficulty of producing satisfactory anatomical or biomechanical representations in the context of open surgery has been an impediment to the development of simulators in this area. In addition, the challenge of developing appropriate physical interfaces to produce haptic feedback has been another limiting factor. Consequently, researchers have focused on “part-task” trainers, in various clinical areas: vascular surgery, dentistry, facial reconstruction, ophthalmology, and cardiac surgery.

In the area of vascular surgery, a training system for microsurgical anastomosis was developed in 1998 through a collaboration between DARPA, Boston Dynamics and Pennsylvania State Hershey Medical Center. The simulator used a spring-mass model to describe the behavior of the vessels, and detachable surgical tools were mounted onto force feedback devices. The haptic devices were used to measure the position of the instruments and to apply interaction forces to the user's hand. Although limited in its ability to characterize the biomechanical behavior vascular structures, the system was visually realistic through the use of advanced rendering techniques.

Few simulations have been done in the area of open heart surgery, at the exception of a training system developed by Mosegaard and Sorensen (Sorensen *et al.*, 2006). The authors have proposed a simulator for complex heart surgery, such as procedures on congenitally malformed hearts. Anatomical heart models were reconstructed from 3D MRI data (Mosegaard *et al.*, 2005) and integrated in graphical settings representing the surgical environment. The simulator uses an underlying spring-mass system to describe the deformable behavior of the heart. By solving the system entirely on the GPU, update rates of about 500Hz are possible, allowing for haptic feedback. Cutting and suturing are also possible, and detailed visual models can be used by decoupling the visualization from the underlying simulation. With a surface description of 135,000 triangles and an underlying spring-mass model of 20,000 nodes, the system achieves visual feedback at 30 Hz.

In the field of brain surgery, the majority of the simulations have focused on pre-operative planning and more particularly the computation of the brain shift that occurs when the skull is opened. Finite element models of the brain, using constitutive laws based on linear and non-linear elasticity, have been used in this context (Wittek *et al.*, 2005), as well as for non-rigid registration of pre-operative three-dimensional medical images of the patient’s brain (Ferrant *et al.*, 2001). None of the models was aimed at interactive simulation, even though computational efficiency was a constraint the different approaches tried to account for.

Ophthalmic surgery

Ophthalmic surgery is an example of a micro-surgical procedure where medical simulation can play an important role. It involves a number of complex procedures that require a high level of training, and excellent dextrous skills. The most frequent and difficult procedures are: cataract surgery, glaucoma surgery, corneal surgery, vitro-retinal surgery and eye muscle surgery. Due to the small size of the eye, instrument motion must be well controlled and the margin for error is very small. Although these reasons are excellent motivations for developing training systems, little work has been done in this area (Neumann

et al., 1998) (Santerre *et al.*, 2006) (Barea *et al.*, 2007). One commercial product exists, the eye surgery simulator EYESi, from the company VRmagic.

Neumann *et al.* (Neumann *et al.*, 1998) described a vitrectomy simulator based on vectorial spring-mass model, i.e. a modified spring-mass model in which forces not only depend on the variation of length of the springs, but also on the variation of angle between spring directions. In the prototype developed by Neumann, user interactions are driven by actual surgical instruments connected to position tracking sensors, allowing tissue manipulation including cutting. No force feedback device is used, which is understandable since limited haptic feedback exists in the real procedure. The simulator developed by Meseure *et al.* (Meseure *et al.*, 1995) targets retinal laser photocoagulation. Designed in close collaboration with ophthalmologists, it includes a large library of cases, to offer an ideal training setup. An actual slit-lamp is supplied with sensors measuring all the degrees of freedom of the system. The images are calculated by a computer and are displayed on a miniature screen placed inside the binocular. A user interface is provided through a Computer Assisted Training software, which allows users to manage pigmentation and pathology data base, to follow the students skill level and to prepare critical operations. Further developments by the same team (Santerre *et al.*, 2006) aimed at cataract surgery. The eye lens was modeled as a linear elastic material using a finite element approach and a spring-mass model was used to describe the lens capsule. The contact with the lens is simulated by a collision sphere, which provides the interaction between the surgical instruments and the virtual environment. The surgical instruments viewed on the screen are controlled by a stylus with six degrees of freedom. The resulting simulation allows basic training for phacoemulsification in real-time with rather realistic visual feedback.

Dental surgery

While the development of computer-assisted oral and maxillofacial surgery systems has been studied in depth, dental surgery simulation remains marginal. Dental surgery simulation is mainly concerned by two aspects of surgery: basic and advanced skills training, and patient-specific planning for dental implants. Historically, preclinical dental instruction was accomplished in a bench-type laboratory environment where students learned psychomotor skills using mannequin heads or hand-held dentiforms (models of maxillary or mandibular plastic jaws with removable plastic teeth). To get away from the traditional laboratory bench technique, more contemporary dental simulation systems were developed in the late 1980s. The majority of these systems use sophisticated mannequins with adjustable settings to create more realistic training scenarios. Although less frequently used, computer-based training systems have made their way into medical schools and training centers. The DentSim[2] device seems the most often cited training system. DentSim is essentially a computer-enhanced patient mannequin. It includes a head and partial torso, dental instruments, infrared sensors on the instruments and mannequin, an overhead infrared camera, a monitor, and two computers. Sets of plastic teeth are inserted into the mannequin's mouth so students can perform drilling and cutting techniques. The sensors allow the computer to make a 3D image showing where the student is drilling and if the work is correct. The machine provides constant, detailed feedback. For added realism, the simulation contains fictional patient records, including medical histories, X-rays, examination and diagnosis notes, and treatment plans.

[2] DenX Ltd, Jerusalem, Israel (denx.com).

Full computer-based systems for dentistry training or planning, on the other hand, are very rarely used and research in this area is limited. The company *Digisens* developed a teaching simulator for dental surgery that lets students refine their surgical gestures on virtual teeth that are accurate and can be patient-specific. Anatomical models are reconstructed from micro-tomographic images, which permits a reconstruction of the surface and internal elements of a tooth. A few of these training systems have been acquired by dental surgery schools but their availability remains limited. On the academic front, most of the work has focused on haptic feedback, since anatomical and biomechanical modeling are less challenging, at least when training is concerned. Haptic feedback is of course of prime importance in dental surgery. In the work of Thomas *et al.* (Thomas *et al.*, 2001) the authors describe the development of a prototype dedicated to the detection of carious lesions, with an emphasis on the technical issues associated with haptic and graphics subsystems. In (Heiland *et al.*, 2004), the authors describe a virtual three-dimensional model of a skull created from Computed Tomography data using the Voxel-Man system. To achieve a realistic drilling effect with the force feedback system, special tools were developed to provide high resolution collision detection. A prototype of dental surgery simulation was developed and evaluated by dental students. In Yau *et al.* (Yau *et al.*, 2006), the authors propose a new approach based on an adaptive octree data structure to precisely compute a distance field of the tooth. This allows for efficient collision detection and haptic force computation. Stable haptic interaction is obtained using a filtered penalty method and the different dental tools are represented with an implicit function for an efficient and accurate calculation of tool-tooth interactions. The authors demonstrate their approach on examples of tooth preparations, such as inlay and bridge preparation.

Biopsy and brachytherapy

Biopsy is a medical procedure widely used in oncology for diagnosis purpose, in clinical areas such as breast or prostate cancer. It consists in introducing a needle through soft tissues to collect samples for further analysis. Brachytherapy is a procedure which is increasingly used for cancer therapy as it is less invasive than surgery and could potentially reduce iatrogenic effects. The brachytherapy procedure consists in implanting radioactive seeds in the prostate using needles. For both brachytherapy and biopsy, the surgical operations are guided by ultrasound images. However, the poor quality of ultrasound images as well as movements and deformations of the tissues make these procedures difficult to perform. This has been a motivation for the development of training and pre-operative planning systems. Marchal *et al.* (Marchal *et al.*, 2006) have proposed a discrete model in order to simulate both external and internal interactions. This model uses a specific formulation for elasticity, based on shape memory, and a particular attention was given to the modeling interactions caused by the needle and the ultrasound probe. Similarly, in (Mohamed *et al.*, 2002) the authors proposed a combination of a statistical and biomechanical model to simulate the deformations of the prostate due to an ultrasound probe. Their model is based on the FEM and is composed of a prostate and its surrounding environment. The influence of surrounding organs geometries is not combined with the study of the ultrasound probe interaction. Modeling biopsy needles and the interaction forces occurring during insertion in soft tissues has been studied by (DiMaio *et al.*, 2003). The authors used a quasi-static finite element method that achieves extremely fast update rates (500Hz) and high accuracy, but requires a calibration phase where the force distribution along the needle shaft is estimated based on observed tissue deformations. Unfortunately, this force distribution, which is modeled with a parameterized surface, may be difficult to measure *in vivo*.

Interventional radiology

Recently, interventional radiology (another important minimally-invasive alternative to surgery) has been the focus of both commercial training systems and research work. The rationale for developing simulators for this type of procedure is similar to laparoscopic surgery. Procedures are rather technical, using indirect vision of the operative field (in this case through X-ray imaging) and flexible devices introduced through the vascular system. Real-time computation of deformable structures, fast collision detection, complex contact processing, advanced rendering techniques, and physiological modeling, are also needed. However, the problems raised are different and in many instances, complementary of the problems related to laparoscopic surgery simulation.

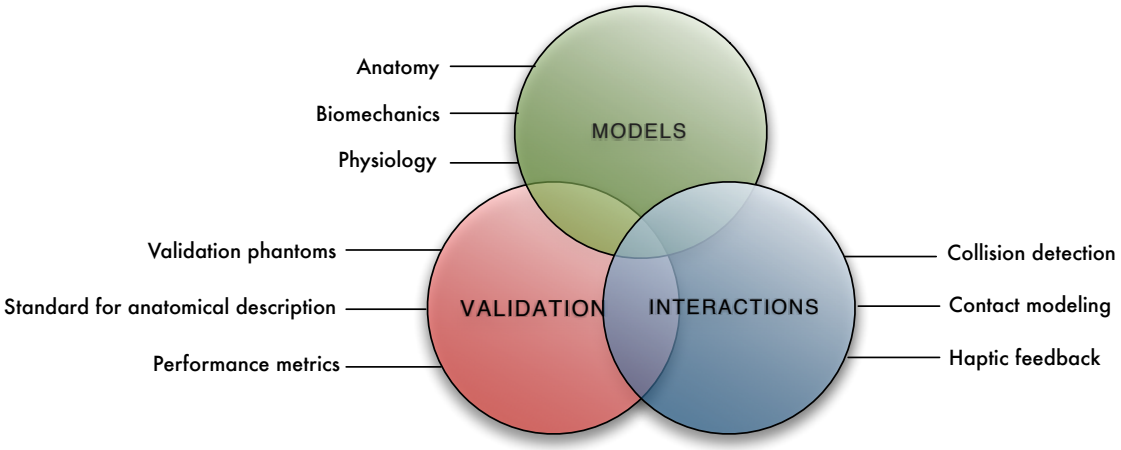
Nowinski *et al.* (Nowinski *et al.*, 2001) developed the NeuroCath system, an interventional neuroradiology pretreatment planning system that provides integrated functionalities for both training and patient-specific planning. In this system the catheter is simulated using a linear finite element model. The visual realism of the simulation is not very convincing, and physiological aspects of the simulation are not described. The interventional cardiology training system ICTS (Dawson *et al.*, 2000) is a computer-based simulator that incorporates synthetic fluoroscopy, real-time three-dimensional interactive anatomic display, and coronary catheterization and angiography using actual catheters. The catheter simulation is based on a multi-body dynamics system. Blood flow distribution in the vascular network is also modeled as simple laminar flow. The system led to the Procedicus VIST system (Mentice) which is among the most successful commercial products in the field of computer-based medical simulation. Yet this system includes a limited representation of the vascular anatomy, and offers a simulation of fluoroscopic images of reduced quality. The Catheter Instruction System CathI (Hoefer *et al.*, 2002) is another computer-based training system which provides force-feedback for guidewire and catheter manipulation. The system is aimed at cardiac procedures and includes a morphologically correct coronary artery model. In (Duriez *et al.*, 2006a) and (Cotin *et al.*, 2007) we proposed a series of solutions for developing a high-fidelity training system for interventional radiology. We particularly focused on the modeling of contacts between flexible devices (catheters, guidewires, coils) and the inner surface of vessels. We also proposed numerous improvements to the modeling of flow and contrast agent propagation. Advanced techniques were also used to improve visual feedback, by relying on volume rendering for instance. A prototype of the simulator, integrating all these different algorithms, was developed in close collaboration with interventional radiologists. A real-time simulation of interventional radiology procedures is described by Wang *et al.* (Wang *et al.*, 2007), which deploys an efficient physics-based thread model to simulate the elastic behavior of guidewires and catheters. A fast collision detection scheme provides continuous collision response, which reveals more details of arterial walls than a centerline approach. The simulation is updated at a rate of 500 Hz to provide haptic feedback.

3. Research in medical simulation

While the list above is certainly not exhaustive, it gives a good overview of the variety of clinical and educational applications of interactive medical simulation, as well as an overview of the variety of approaches used by researchers to address the different issues raised by these very demanding applications. Overall, most difficult problems in medical simulation lie at the intersection between the research areas we described previously: biomechanics, interaction, and real-time computation. However,

there exist other important elements to account for, such as curriculum development or performance metrics, for instance, which have been less studied in the field of computer-based simulation. Finally, systems integration and validation are other essential elements in medical simulation, as they are permit to assess, both qualitatively and quantitatively, our work.

The following sections describe my principal contributions in various areas of medical simulation. In particular, I describe through several examples the importance of anatomical modeling as a mean to create realistic, patient-specific, virtual representations of the operating field. I also address the challenge of determining soft tissue characteristics, and defining mathematical models of soft tissue deformation. In this context also, the constraint of real-time computation requires to simplify, adapt, and optimize the models to enable real-time computation. It becomes therefore essential to develop validation mechanisms to determine the range of validity of the simplified models and to qualitatively compare various approaches against each other. A series of contributions related to the modeling of tissue-tool interactions is also presented. Whether rigid instruments are used to manipulate soft tissues or flexible devices navigate inside anatomical structures, this interaction is an essential part of a medical procedure. Modeling such interactions, through efficient collision detection, and advanced contact mechanics, remains a challenge.



The following chapters (Chapter II, Chapter III, and Chapter IV) are organized around three main themes: Models, Interaction, and Validation. Since real-time computation is an inherent constraint to each of these topics, it is addressed specifically through each theme. Dependencies between Models, Interaction, and Validation, as illustrated by the diagram above, will be addressed in Chapter V through a series of examples of simulation systems we have developed during our research. Finally, a description of future directions for our work, as well as a proposition for a Open Source framework for developing this work, will be highlighted in Chapters VI and VII.

Chapter II — Models

1. Introduction

A model is a representation of the essential characteristics of an object or event in the real world. This representation usually takes one of the following forms: physical (e.g. an architect's model of a building) or symbolic (e.g. a set of mathematical equations). In either form, models present many advantages over the real world. They are easier to manipulate (by changing parameters of the model) and analyze than real world objects. However, models are necessarily incomplete: no model includes every aspect of the real world. If it did, it would no longer be a model.

To create a model, assumptions must be made about the essential structure and relationships of objects in the real world. These assumptions should be related to what characteristics are necessary to explain a particular phenomenon. In the context of medical simulation, it is essential to define the scope and objectives of the simulation in order to make such assumptions. A simulator designed for teaching basic laparoscopic skills may not need an advanced biomechanical soft tissue model. In the example of the MIST system (Seymour, 2002), or the CELTS system (Maithel, 2006), training efficiency has been proven although no accurate modeling of the anatomy or the soft tissues was provided. On the other hand, for more advanced training, as well as procedure planning and rehearsal, it is essential to model as accurately as possible every aspect of the operating field. For interactive simulation, a key requirement is to guarantee real-time computation. This constraint greatly limits the range of biomechanical models that can be used, yet it should not be mixed with the initial modeling hypotheses. Assumptions should relate to the main characteristics of the object we want to translate into the model, while the real-time constraint should be taken into consideration once a model is determined. These two steps are often mixed, but should probably not. It is important to first characterize the real world in order to have a reference against which optimized, real-time models can be compared.

In the following sections are presented various aspects of modeling as well as the specific algorithms that have been developed to address the allow interactive simulations. Below is an overview of the different areas that will be discussed.

- **Anatomy**
  - **Generic and patient-specific anatomical modeling**
  - **Models suitable for real-time (interactive) computation**
- **Biomechanics**
  - **Measuring soft-tissue properties**
  - **Constitutive models and parameter identification**
  - **Models suitable for real-time (interactive) computation**
- **Physiology**
  - **Vascular physiology**
  - **Models suitable for real-time (interactive) computation**
- **Medical devices**
  - **Modeling flexible medical devices**
  - **Modeling visualization devices**

2. Anatomical Modeling

With the exception of specific training systems, such as laparoscopic skills trainers where abstract models of the anatomy can be sufficient, most training systems require to model the geometry of the operating field with accuracy. Anatomical representations often correspond to a generic “patient”, in a similar way as an anatomy book describes human anatomy. While such a representation of the anatomy – often based on an average model – has value for training new procedures, it remains important to account for shape variability for more advanced training. An anatomical database, corresponding to various pathologies, can then be created and integrated with appropriate training scenarios. Finally, in the case of planning or rehearsal of complex procedures, patient-specific models of the anatomy need to be created, with the highest possible accuracy and in an automated manner.

Whether we develop generic or patient-specific simulation systems, creating anatomical models raises two challenges. First, there exists no image processing technique that can automatically generate an accurate model of an organ or anatomical structure (Pham *et al.*, 2000). The best approaches are still semi-automatic and remain highly contingent on the initial image quality. However, with constant improvements in image acquisition techniques, the potential for creating accurate models of the anatomy increases every day. The second challenge in creating anatomical models comes from the specific requirements of real-time computer-based simulation. Not only the models need to be rendered at interactive rates, but they also need to exhibit additional properties, required by other components of the simulator. For instance, the surface mesh of the reconstructed structure is likely to be used as a basis for creating a volumetric mesh for finite element analysis. However, finite element meshes need to meet certain criteria to ensure a well-conditioned “stiffness” matrix and therefore optimal convergence rates. For collision detection, other criteria apply, such as minimizing the number of primitives describing the surface of the “object” while preserving the main characteristics and topology of the surface. Also, when considering advanced collision response between soft-tissue and medical devices, smoothness of the surface will play an important role in the computation of the response.

Although medical simulation does not directly address the first challenge of surface reconstruction of anatomical structures from medical images, it needs to address the second one. Below we provide elements of response for this problem, through two key contributions. The first one presents an approach for creating surface models of organs which exhibit good properties for further computation using a finite element method. The second contribution deals with a complex problem: a topologically and geometrically sound representation of vascular networks, optimized for collision detection and flow computation.

Anatomical Modeling of the liver

In the method proposed in (Cotin *et al.*, 1996a), we created an optimized liver model through an accurate segmentation of a volumetric medical image, followed by an appropriate representation of the segmented organ’s surface. In order to produce a model of the liver with anatomical details, a data set provided by the National Library of Medicine was used. This data set consists of axial MRI images of the head and neck and longitudinal sections of the rest of the body. The CT data consists of axial scans of the entire body taken at 1 mm intervals. The axial anatomical images are high resolution scanned pictures of cryogenic slices of the body also taken at 1 mm interval and coinciding with the CT axial images. To

extract the shape of the liver from this data set, we used the anatomical slices for their better contrast between the liver and the surrounding organs. Of the entire initial data set, the data directly related to the liver anatomy consists of about 180 slices. After contrast enhancement, an edge detection algorithm was applied to extract the contours of the image, and through a thresholding technique we retained the stronger ones. Next, we used semi-automatic deformable models for two-dimensional contour extraction to generate a set of two-dimensional binary images (cf. Figure 1). The slices generated by this process were then stacked to form a three-dimensional binary image (cf. Figure 2). Usually, a smoothing of the volume data is necessary to reduce staircase effects (cf. Figure 2), then a mesh of the surface is usually obtained through an algorithm such as the Marching Cubes (Lorensen *et al.*, 1987) but the resulting mesh generally contains too many triangles to be processed in real-time. Although it is possible to rely on surface decimation tools to reduce the number of geometric primitives, the resulting surface usually lacks some of the properties needed for further processing. To remedy this issue and introduce more flexibility in the reconstruction process, we proposed to use *simplex meshes*.

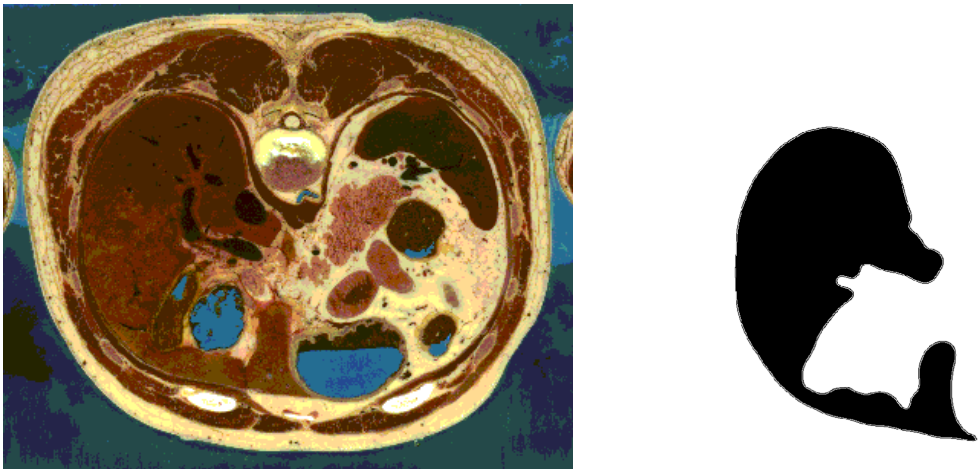


***Figure 1**: Segmentation of the liver, slice by slice, The initial data (left) is a high resolution photography of an anatomical slice of the abdomen. The binary image (right) corresponds to the segmented liver cross-section.*

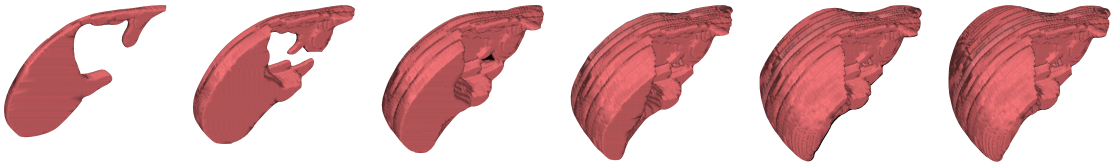


***Figure 2**: After segmentation, the binary images are stacked (from left to right) to give a three-dimensional binary image. On the final volume (right) we can see the step effect on the shape of the liver. The resulting surface that can be generated from such volumetric segmented data is generally not smooth and contains too many geometric primitives for real-time processing.*

Simplex meshes are an original representation of three-dimensional objects developed by Delingette (Delingette, 1997). A simplex mesh is an active surface that can be seen as a three-dimensional extension of an active contour or “snake” (Kass *et al.*, 1987) (Sundaramoorthi *et al.*, 2006) and therefore is well suited for generating geometric models from volumetric data. A simplex mesh can be deformed under the

action of regularizing and external forces. Additional properties, inherent to this representation, are the constant connectivity between vertices and the existence of a duality with triangulations. Simplex meshes are also adaptive. By concentrating vertices in areas of high curvature, they can lead to optimal shape descriptions for a given number of vertices. The mesh may be refined or decimated depending on the distance of the vertices from the data set. Figure 3 shows the effect of the mesh adaptation, in particular the concentration of vertices around areas of high curvature. By integrating simplex meshes into the segmentation process, we have obtained smoothed triangulated surfaces very close to what could be obtained using a Marching Cube algorithms, but with fewer faces to represent the shape of the organ. In our example, the model of the liver has been created by fitting a simplex mesh to the tridimensional binary image previously described. Thanks to the adaptation and decimation properties of the simplex meshes, this resulting model comprises only 14,000 triangles (see Figure 3), whereas the marching cubes algorithm gave 94,000 triangles. Although this approach was useful for building a "generic" liver model, it can also be used to create "patient-specific" models. Montagnat *et al.* (Montagnat *et al.*, 1997) proposed a method for extracting liver models from CT scan images using simplex meshes with additional shape constraints, making the method more robust while maintaining the same benefits as the model described here.

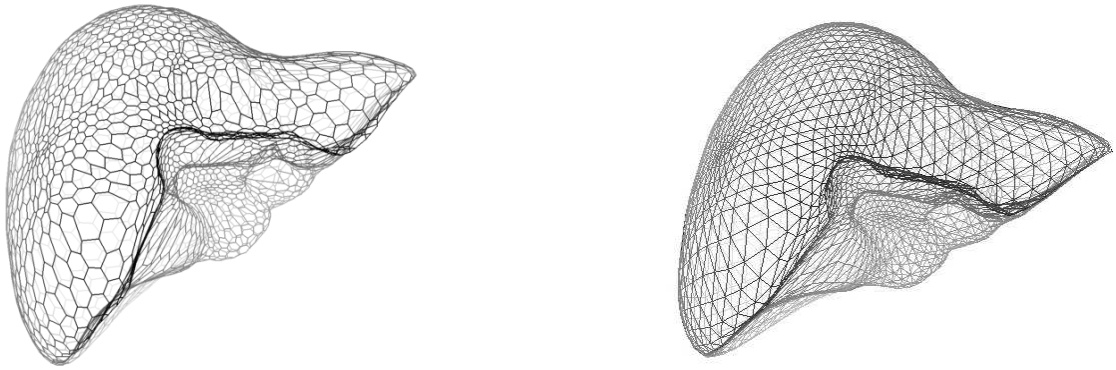


***Figure 3**: Different representations of the surface mesh of a liver. The simplex mesh (left) with a concentration of vertices in areas of high curvature, and the triangulated dual surface (right).*

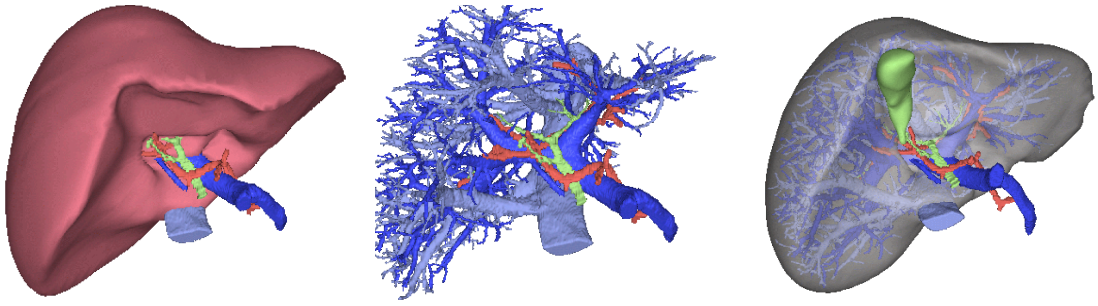


***Figure 4**: Different views of the anatomical model of the liver. The complete model (left and right) and the vascular network (middle) also segmented from the anatomical color slices of the Visible Human project.*

Once an optimal surface mesh of the organ is obtained, most deformation algorithms require the definition of a volumetric mesh as a support for the computational model. In the case of the liver, we used a linearized formulation of three-dimensional elasticity equations (Cotin *et al.*, 1996a). To solve these equations we proposed a modified finite element approach (Cotin *et al.*, 1999). This approach requires the creation of a volumetric mesh of the liver composed, in our case, of tetrahedral elements. While tetrahedral elements may not be optimal in terms of computational performance, they are often a preferred choice for the purpose of meshing irregular shapes such as anatomical structures. Thanks to the duality property of simplex meshes, creating a triangulation of the liver surface is trivial. The volume was then meshed into a set of tetrahedra without any modification of the original triangulation. A commercial product, Simail(TM), which is based on a Delaunay-Voronoi algorithm, was used for volumetric meshing. The resulting mesh, which contains about 10 times less elements than using a conventional approach, allows real-time computation of the deformations while preserving enough anatomical details for realistic training (see Figure 5).

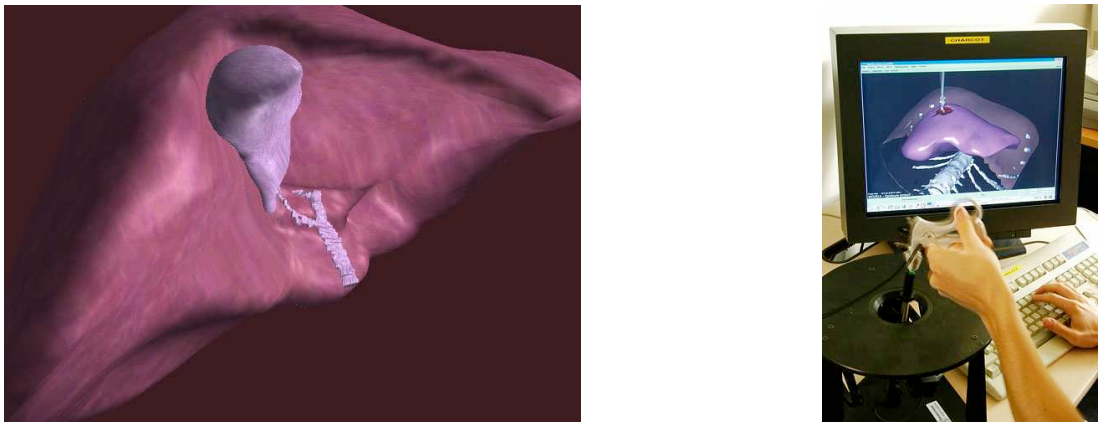


***Figure 5**: Texture-mapped model of the reconstructed liver and gall bladder. Liver model used in an interactive simulation of laparoscopic surgery.*

Anatomical modeling of vascular systems

In the context of endovascular therapy, and particularly for interventional radiology, anatomical modeling also raises several challenges. From a segmentation and reconstruction standpoint, vascular networks are among the most complex structures to segment: the diameter of most vessels is only a few voxels wide, and the proximity between vessels makes it difficult to create a topologically correct representation. In addition, when real-time simulation or visualization is considered, additional constraints need to be taken into account during the reconstruction process. In (Luboz *et al.*, 2005) we have proposed a method for generating smooth geometrical representations of vascular networks while topologically preserving two-manifold surfaces. The reconstructed surface is well-suited for further processing needed by the simulation such as efficient collision detection and response between the vascular surface and the virtual medical devices (Cotin *et al.*, 2005), physical-based blood flow simulation (Wu *et al.*, 2007), and visualization. We describe hereafter the two main steps of the method: vessel segmentation and three-dimensional reconstruction. It is important to note that similarly to the approach presented above for the creation of an anatomical model of the liver, the proposed method integrates constraints inherent to real-time simulation in the reconstruction process itself, rather than through a post-processing stage. This is a

critical point, specially for vascular models, since post-processing such as surface simplification often leads to non-desirable geometrical outcomes (such as the reduction of the diameter of the vessels during the surface decimation process).

Segmentation and reconstruction of vascular networks: different techniques can be used for the segmentation of vascular structures. The simplest and sometimes more efficient technique consists in thresholding, although it usually requires preprocessing the data to identify small tubular structures. More complex methods, but also with a higher potential for computing a suitable topological representation of the vascular network, can be divided in two main approaches: a) Techniques for centerline enhancement, including multi-scale approaches, usually based on the Hessian matrix, or by applying a thinning technique, such as a homotopic skeletonization; b) techniques for contour extraction, including statistical approaches such as Expectation Maximization (Wells, *et al.*, 1996) or random Markov fields, and geometrical approaches, such as region growing, adaptive thresholding, and active contours. Active contours can have an explicit formulation, such as snakes (Kass *et al.*, 1987) (Sundaramoorthi *et al.*, 2006), or implicit, such as level sets (Sethian *et al.*, 1999). Both techniques are very efficient to segment large vessels such as the aorta or the carotids. However, they are not well suited for accurately segmenting small vessels, especially in the brain. In all cases, and whatever the chosen segmentation technique, it is usually preferable to apply a noise reduction algorithm, such as an anisotropic diffusion filter, before segmenting the vessels.

The method we proposed in (Luboz *et al.*, 2005) is a segmentation technique composed of the following steps: 1) segmentation of the vessels through a level set evolution, initialized using an adaptive threshold; 2) skeletonization to extract centerlines and cross-sections of the vessels; 3) surface reconstruction. Below is a brief description of the process involved in each of these steps:

Segmentation: to improve the robustness of the segmentation process, in particular when using Computed Tomography Angiography (CTA) data sets, the data is pre-processed and voxels representing bones, sinuses and skin are removed, as they have a similar intensity as the one of the vessels filled with contrast agent. A set of masks, based on morphological operations, is used to perform this task. Then, to reduce noise in the data while preserving small vascular structures, an anisotropic filter (Krissian, 2002) is used. This enhancement improves the segmentation of small vessels, such as vessels of diameter inferior to 2.0 mm. Such small vessels are numerous in the vascular system of the brain. Vessel contours are extracted by means of a level set evolution (Osher *et al.*, 1988) (Sethian *et al.*, 1999) applied on the enhanced data set. Although the level set technique is computationally more expensive than simple thresholding, it allows a better estimation of the contours. Our implementation for fast evolution of the contour is based on work done by (Krissian *et al.*, 2005). The level set evolves a surface according to three different forces: an advection force that pushes the surface towards the edges of the image; a smoothing term, proportional to the minimal curvature of the surface, that keeps the surface smooth; and a balloon force F that allows the contours to expand within vascular structures

$$F = e^{-\frac{(I-m)^2}{\sigma^2}} - \tau \qquad (1)$$

These forces rely on intensity statistics to either expand or shrink the evolving contour. I is the intensity, m stands for the mean intensity of the vessels, σ is their standard deviation, and τ is a threshold (0.2 by default) that allows shrinking the contour when the position is unlikely to belong to a

vessel. For this study, a 3D surface of the vessels is reconstructed using a marching-cubes algorithm with an isosurface of intensity zero from the result of the level set evolution.

Skeletonization: from the binary images segmented by the level set method, a skeletonization technique is applied to extract the vessels centerlines. Extraction of the centerlines from the segmented data set is based on homotopic thinning (which preserves the topology of the vascular network) where voxels are iteratively removed based on their Euclidean distance to the segmented surface. Extracting centerlines and local radius information rather than creating a surface mesh as in the previous example was a choice governed by other computational aspects of the simulation. In particular, the existence of an abstract topological description (the centerlines) of the vascular network is necessary to compute blood flow for instance. It also offers a convenient way to index local properties of the vessels (such as list of triangles describing the local geometry, set of particles describing the local volume, etc.) as described in (Luboz *et al.*, 2005) and (Cotin *et al.*, 2005). In addition, if the centerlines are connected, i.e. if they describe a graph, it is possible to reconstruct the complete vascular network without creating any hole, which would be very difficult using a surface extraction such as the marching cube algorithm (Lorensen *et al.*, 1987). To enable accurate vessel reconstruction using this centerline description, an estimation of the local cross-section of the vessel is needed. While representing cross-sections as circles already permits a good approximation of the overall shape of the vessels (Luboz *et al.*, 2005), more robust fitting and more accurate shape description can be obtained by fitting an ellipse instead of a circle. The ellipse fitting is based on the method described in (Fitzgibbon *et al.*, 1999). Starting from the estimated centerlines and a rough segmentation of the vessel boundaries, ellipses are fitted in the planes of the vascular cross-sections, defined as the planes orthogonal to the centerlines. The ellipses are fitted at points regularly distributed along each centerline (see Figure 6). The process is driven by computing the intensity gradient, i.e. the derivatives of a Gaussian kernel with a given standard deviation. An iterative scheme is added to improve the accuracy of the ellipse fitting algorithm. As ellipses are not rotationally invariant around the axis of the centerline, a better fit can be obtained. This also improves the accuracy of the location of the centerline. Sub-voxel accuracy can therefore be obtained in only a few iterations. Using this approach, errors as low as 0.1 mm have been measured on synthetic data.

Surface reconstruction: to generate an efficient, structured, and adaptive representation of vascular structures from centerline and cross-section data, we have developed a robust and accurate reconstruction method based on the work from (Felkel *et al.*, 2004). Other surface reconstruction techniques, described in a survey by Bühler *et al.* (Bühler *et al.*, 2002) were also considered. However, the method described by (Felkel *et al.*, 2004) was the closest to our initial requirements. Basically, our method reconstructs topologically preserving quadrilateral surface patches of branching tubular structure. These quadrilateral patches can then be subdivided into smaller polygons or converted into spline surfaces. The method we proposed for generating a surface mesh from centerline and cross-section data relies on the creation of a series of generalized cylinders (with circular or elliptical cross sections). Each vessel is discretized into a set of segments while cross sections are initially approximated by four-sided polygons. The difficulty is then to correctly patch the various patterns of branching vessels that can appear in complex vascular networks. Our main contribution was to improve the smoothness of the reconstructed vascular structure with less patching artifact while preserving branching symmetry. These are important factors for improving the quality of any simulation based on such reconstructed models. In particular, the reconstruction scheme we proposed takes into account both branching angle and vessel cross-section area to reduce under-sampling artifacts and improve the reconstruction robustness.

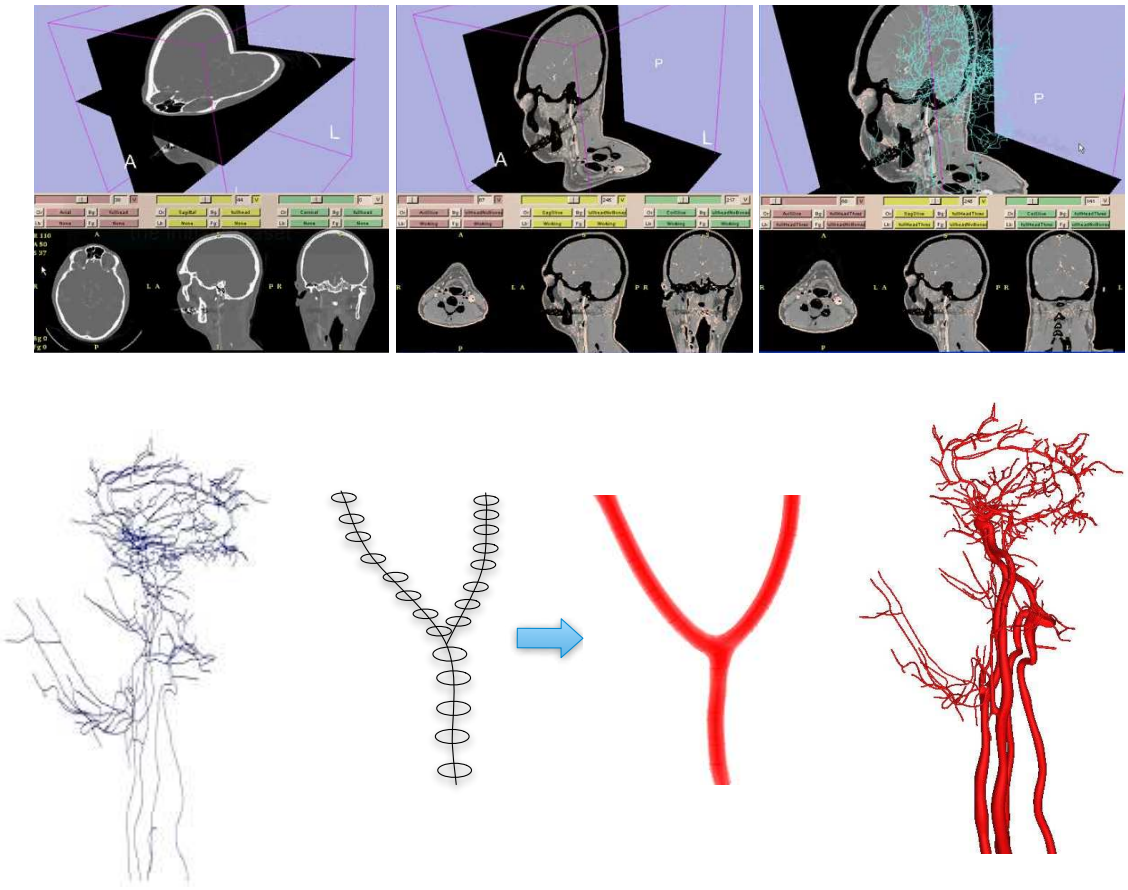


***Figure 6**: Top row - From left to right: initial CTA data set before any pre-processing, pre-processed data set after noise filtering and removal of voxels corresponding to skin and bone tissue, overlaid rendering of the skeleton of the arterial vascular network. Bottom row - from left to right: skeleton generated automatically from the patient CTA, diagram of the reconstruction process, and 3D surface reconstructed from the skeleton and cross-section data.*

A new bifurcation tiling with end-segment-grouping and adjacent-quadrant-grouping was also proposed to reduce surface patching artifacts when the branching angle is close to 0 or 90 degrees and to reduce unwanted surface twist. Final surface-fitting is answered by a Catmull-Clark or Loop subdivision algorithm (Biermann *et al.*, 2000) on the coarse mesh to improve surface smoothness. At each subdivision the number of triangles in the surface mesh increases by a factor 4, but only two levels of subdivision are typically needed to achieve excellent results (see Figure 7) since at each subdivision the cross-section is approximated more and more closely, thus giving an overall more accurate surface description. The subdivision also naturally leads to a smoother surface, which is key for simulating the navigation of medical devices.

Anatomical model of the cerebrovascular system: the method described above has been applied to various data sets, in particular contrast-enhanced magnetic resonance images (MRA) and contrast-enhanced computed tomography images (CTA). With MRA data, it is possible to obtain a fully automatic segmentation and reconstruction of the main cerebral vessels. The skeleton obtained from such data contains enough details for learning the basics of diagnostic angiography through catheter navigation and contrast agent injection. Using a higher resolution CTA data set, with manual interactions, it was possible

to accurately segment and reconstruct a very detailed model of the vascular system, with about 3,000 arteries and 3,000 veins. Figure 7 shows the skeleton and the reconstructed surface.

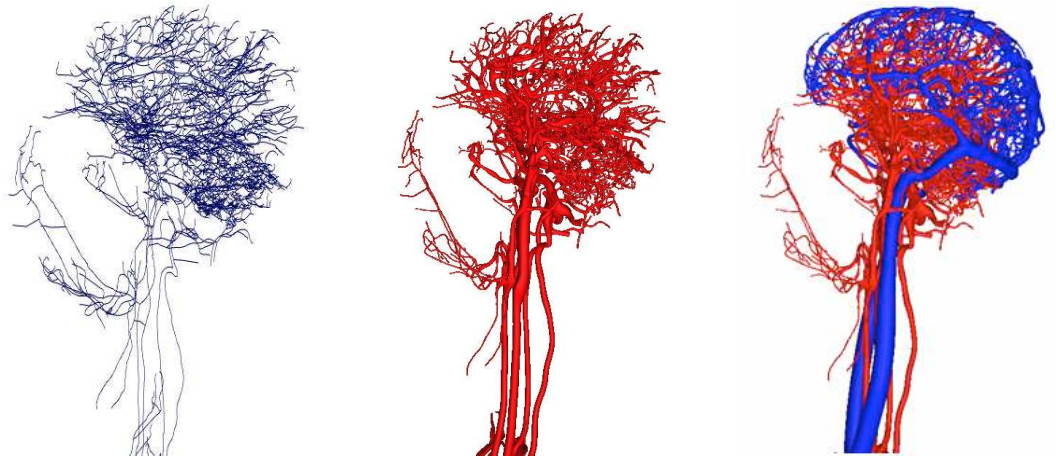


***Figure 7**: From left to right: (a) skeleton generated semi-automatically from the patient CTA, (b) 3D vascular surface reconstructed from the skeleton and cross-section data, and (c) combined arterial + venous sides obtained from the same patient data set. Arteries are in red, veins in blue. One can see the complexity and level of detail of the reconstructed models, and understand the importance to have optimized representations for real-time processing during a simulation.*

The skeleton has 3,388 lines and the reconstructed surface 58,612 polygons. This level of details is often required to allow an accurate diagnosis of the smallest brain vessels. From the same CTA, the venous system has also been reconstructed using our method. The resulting skeleton has 1,660 lines and the reconstructed surface 28,322 polygons. The combination of the arterial side and the venous side, shown in Figure 7 provides an ideal anatomical model for learning all essential steps of a procedure.

Anatomical model of the cardiovascular system: we have applied our method to a CTA of the chest in order to segment the vascular system of the heart (coronary arteries and veins) as shown in Figure 8. The skeletons contain 79 lines and 35 lines for respectively the arterial side and the venous side.

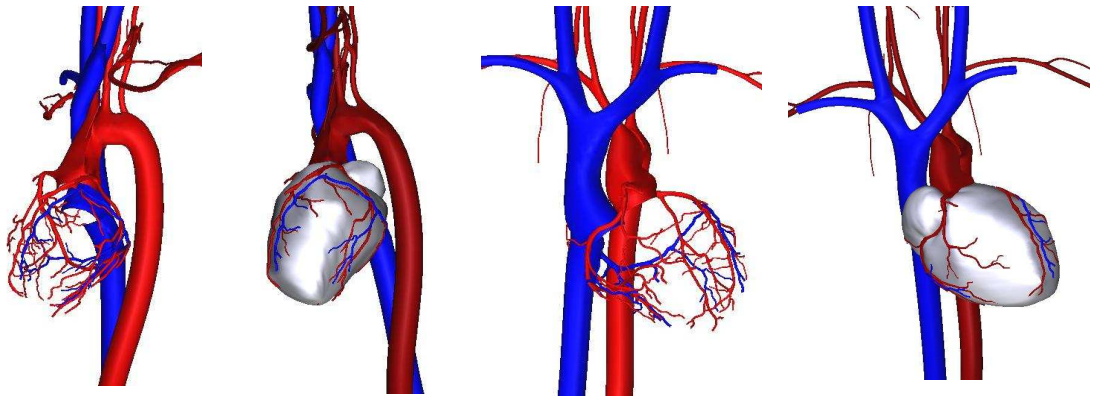


***Figure 8**: Coronary arteries and aortic arch reconstructed from a CTA data set. Both the arterial and the venous sides are displayed (arteries in red, veins in blue).*

The robustness and the accuracy of the method have been quantified on the vascular phantom mentioned above and compared to a surface generated using a marching cube algorithm and a combination of decimation and smoothing methods. The results showed that our approach leads to very accurate models of vascular networks, while permitting real-time simulation since the surface contains a minimal number of geometric primitives and yields to smooth surfaces, ideal for collision and contact modeling.

Future directions

Anatomical modeling plays a key role in medical simulation. Not only does it permit to recreate a meaningful clinical environment from patient-specific data, but it also serves as a basis for many other processing aspects of a simulation. Currently, the best way of creating multiple representations of the anatomy suited for each computational requirement is to perform a post-processing of an initial mesh to obtain (usually coarser) meshes suited for collision detection or finite element modeling. Yet, such an approach does not always yield ideal results, as the initial anatomical model might not have the right characteristics required by other simulation components. The work presented in this section and illustrated with two very different cases, is a step in a different direction. I believe that reconstruction techniques from medical images need to be adapted to the particular requirements of medical simulation. Currently, most of the work focuses on producing visually detailed anatomical models. We need to extend this work, as we have started to do, to integrate constraints specific to our problems directly into the reconstruction process.

Summary of contributions

- **Research articles**
  - X. Wu, J. Allard, and S. Cotin. “Real-time Modeling of Vascular Flow for Angiography Simulation”. Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), Volume 4792, pp. 850-857, 2007.
  - J. Rabinov, S Cotin, J. Allard, J. Dequidt, J. Lenoir, V. Luboz , P. Neumann, X. Wu , S. Dawson. “EVE: Computer Based Endovascular Training System for Neuroradiolgy”. ASNR 45th Annual Meeting & NER Foundation Symposium, pages 147--150, 2007.
  - V. Luboz, X. Wu, K. Krissian, C.-F. Westin, R. Kikinis, S. Cotin, S. Dawson. “A segmentation and reconstruction technique for 3D vascular structures”. Proceedings of the MICCAI Conference, MICCAI 2005, pp 43-50, 2005.
  - Wu X, Krissian K, Luboz V, Cotin S, Dawson S. Smooth Vasculature Reconstruction from Patient Scan. Proceedings of the Virtual Environment Interactions and Physical Simulation workshop, VRIPHYS, Pisa, Italy, 2005.
  - X. Wu, V. Pegoraro, V. Luboz, P. Neumann, R. Bardsley, S. Dawson, S. Cotin. “New Approaches to Computer-based Interventional Neuroradiology Training”. Proceedings of 13th Annual Meeting, Medicine Meets Virtual Reality; January 2005.
  - P. Neumann, S. Cotin, X. Wu, V. Pegoraro, V. Luboz, and S. Dawson. “High-Fidelity Interventional Neuroradiology Training System”. Proceedings of the 30th Annual Society of Interventional Radiology Annual Scientic Meeting, April, 2005.
- **Patents**
  - Luboz V, Wu X, Krissian K, Cotin S, inventors; “ADAM: a segmentation and reconstruction method for endovascular and endoluminal anatomical structures”, Massachusetts General Hospital, Provisional patent application.
- **Software**
  - **ADAM:** semi-automatic segmentation and reconstruction tool for vascular structures. This software can extract vascular information from various imaging modalities (CTA or MRA) that can be used in a simulation system. Information consists in a topological description, a description of the vessels centerlines, and a surface description. A volumetric representation of the lumen of the vessel, using multi-resolution voxelization, can also be created.
- **Miscellaneous**
  - Best poster award at the 30th Annual Massachusetts General Hospital Research Symposium.

3. Biomechanical Modeling

Biomechanics is the research and analysis of the mechanics of living organisms. This research and analysis can be performed at multiple levels, from the molecular level all the way up to the tissue and organ level. Medical simulation relates to various aspects of biomechanics, essentially at the organ level, and considers mainly bones, muscles, soft tissues and vessels. Mechanical properties of soft tissues depend not only upon the inherent properties of its constituents but also upon how the constituents are arranged relative to each other. The complex arrangement of the constituents leads to material properties that are often non linear, non homogenous, and non isotropic. Thus, the mathematics necessary to describe soft tissue deformations is much more complicated than with most materials. Consequently, soft-tissue modeling is a very important challenge for interactive medical simulation. Developing algorithms that allow the computation, in real-time, of soft-tissues constitutive laws is very difficult. However, to even start dealing with this problem, another challenge consists in determining the actual characteristics of the tissue or organ to be simulated, and to choose, among the possible models, the one that will ultimately offer the best trade-off between biomechanical realism and computational efficiency. Those various constraints require to 1) determine what are the key characteristics of the tissue that are relevant for the objective of the simulation, 2) perform a series of experiments to determine both a constitutive model and parameters of this model, 3) develop simplified or optimized versions of the model suitable for real-time computation.

All living tissues have numerous constituents, each of which may have distinctive mechanical properties. For example, elastin fibers give some tissues (such as blood vessel walls) their spring-like quality at lower loads; inextensible collagen fibers that are initially wavy and unable to bear much load become straightened to bear almost all of the higher loads; and muscle fibers contract and relax to dramatically change their properties from moment to moment. Interconnecting all these fibers are fluids, proteins, and other materials that contribute to the mechanical properties of the tissue. Below is a simplified overview of the main characteristics of different anatomical structures or tissues that could be represented in a simulator.

Biomechanics of bones: bones are anisotropic but are approximately transversely isotropic, i.e. they are stronger along longitudinal axis than the transverse axis. The stress-strain relations of bones can be approximated with good accuracy using a linear elastic law (Hooke's Law). Modeling bone deformation and fracture is obviously essential for the simulation of orthopedic procedures, but it is also relevant for cranio-facial surgery and procedures where access to the organ requires cutting or drilling through bone (such as brain surgery).

Biomechanics of muscles: there are three main types of muscles: smooth muscles, skeletal muscles and cardiac muscles. Smooth muscle tissue is a complex structure, which lines the hollow internal organs (such as the stomach or the digestive tract), and exhibits non-linear, viscoelastic, anisotropic, time dependent properties. Predicting the stress-strain behavior of such tissue is difficult, and usually relies of three-dimensional finite element simulations (Pidaparti *et al.*, 2003). Hyper-elastic models have also been proposed to simulate the shortening-dependent stiffness of tracheal smooth muscle tissue through three-dimensional non-linear finite element analysis (Thurairajah *et al.*, 2000).

Skeletal muscles have a complex hierarchical structure in which muscle fibers take care of the contraction of the whole muscle. Muscle tissues can be voluntary controlled, and their behavior depend

largely of the frequency and history of the stimulation. Hill's Model is the most popular model used to study muscle. Stress in the muscle is assumed to result from the superposition of a passive and an active part. The passive properties are often described by a hyper-elastic constitutive material law whereas the active part depends on the fiber length, shortening velocity and an activation function (REF). Finite element modeling is usually the preferred approach for solving the rather complex constitutive equations of muscles.

The main characteristic of the cardiac muscles lies in the existence of cardiomyocytes, specialized cells that involuntarily contract and are the source of heart beats. Although the cardiac muscle is composed of separate cardiac cell units, the entire myocardium behaves as a single coordinated contractile unit. Of prime importance in the cardiac function, Purkinje fibers are modified cardiac muscle cells located in the bundle of His, which are specialized for conduction. Since 1960, the physiology of cardiac cells has been the focus of many mathematical models. The first cardiac cell models were those of Fitzhugh (Fitzhugh *et al.*, 1961) and Noble (Noble *et al.*, 1962) which simulated action potentials in Purkinje fibers. With new experimental data, these models were updated later by McAllister, Noble and Tsien (McAllister *et al.*, 1975). Other models have been published, based on new experimental data, to further improve the previous ones. Yet, these models remain very complex and lead to computation times far from real-time. A simplified model was presented in (Sermesant *et al.*, 2006) simulates the electrical and mechanical activity of the ventricles. The transmembrane potential propagation is simulated using FitzHugh-Nagumo reaction-diffusion model and the myocardium contraction is modeled through a constitutive law including an electromechanical coupling. This model enables the introduction of pathologies and the simulation of electrophysiology interventions.

Biomechanics of soft tissues: soft tissues such as tendon, ligament and cartilage are combinations of matrix proteins and fluid. In each of these tissues, the main strength bearing element is collagen, but the amount and type of collagen varies according to the function performed by each tissue. Elastin is also a major load-bearing constituent within skin, the vasculature, and connective tissues. Similarly, the biomechanics of organs and organ systems stem from the various characteristics of their component cells and extracellular materials, which vary widely in structure and composition and hence in biomechanical properties. The cardiovascular system is a good illustration of this complexity, where the heart, blood vessels, and blood all contribute to the overall behavior of the system. Recently, there has been a growing interest in measuring and modeling the biomechanics of soft tissues, in particular for internal organs. See for instance the work of (Ottensmeyer, *et al.*, 2001), (Miller, *et al.*, 2005), (Kerdok, 2006) and (Hollenstein, *et al.* 2006). These various studies show the importance of performing measurements *in vivo* or in conditions close to *in vivo*. They also demonstrate the influence of the different constituents of an organ in its overall behavior. This behavior is approximated by a constitutive model, which determines the state of stress at any point in the tissue in response to any arbitrary history of motion (Holzapfel, *et al.*, 2000). Knowing the structure and the constitutive relationships that describe the material behavior of a composite body (i.e. an intact organ), the response of the body to any prescribed loading condition (i.e. surgical manipulations) can theoretically be obtained. The challenge is to define constitutive relations that are both simple enough to be implemented in real-time and realistic enough to accurately describe the material's behavior across different loading conditions. Currently, the lack of elegant and robust constitutive relations for soft biological tissues hinders the development of predictive biomechanics necessary for medical simulation (Fung, 1993).

Humphrey proposes a five-step procedure for developing a constitutive relation (Humphrey, 2002):

1. Determine the general characteristics of interest
2. Establish a theoretical framework
3. Identify the specific functional forms of the constitutive relations
4. Calculate the values of the material parameters
5. Evaluate the predictive capability of the final relation

The contributions described in the following sections are organized around three main topics, covering the different steps described above: create tools and techniques to interrogate the mechanical responses of soft tissues *in vitro* and *in vivo*; develop mathematical models to capture the observed tissue responses, determine the values of the characteristic parameters for the tissues in question and develop algorithms for real-time computation of the models; and finally validate the instruments, techniques, and models.

Characterizing soft tissues

Measuring tissue characteristics *in vivo* is essential as the properties of non-living tissues are known to be significantly different from those of living organisms, due to the temperature and strain state differences, and the absence of blood perfusion, among other factors. More recently, large scale viscoelastic deformations experiments (Kerdok *et al.*, 2005) have quantified the important difference between tissue properties *in vitro* and *in vivo*, thus questioning the validity of published tissue properties based on *in vitro* experiments. With little research done on *in vivo* soft tissue properties, we currently rely mostly on generic models, such as linear or non-linear elastic models. However, developing experiments and devices to measure tissue properties *in vivo* is very complex, and variability in organ shapes and pathologies makes it difficult to validate the results.

In this section is described elements of a research program that was conducted through a collaboration with the Biorobotics Laboratory at Harvard University, which principal aim was the parallel development of measurement tools and mathematical models. While my contribution to some of the elements presented here was limited, I decided to include them nonetheless in order to have a more complete overview of the complex problem of modeling soft tissue for interactive simulation.

Measurement devices

Several devices were designed to measure *in vivo* and *in vitro* tissue properties of abdominal organs in animals, and in some specific instances, in patients. The TeMPeST 1-D (Tissue Material Property Sampling Tool) is a minimally invasive instrument which imposes small amplitude vibrations to peritoneal solid organs and measures the reaction forces (see Figure 9). By fitting force-displacement responses to the form of a chosen model, the tissue parameters can be obtained. The data acquisition tool is suitable for minimally invasive or open surgical use (Ottensmeyer, 2001). It measures normal indentation force-displacement response over a frequency range from DC to approximately 100Hz. This permits the investigation of the visco-elastic properties of living solid organ tissue. It can exert forces up to 300 mN, and has a range of motion of ±500µm. The TeMPeST 1-D has been used to measure the frequency-dependent stiffness of various soft-tissues in animals *in vivo*.

The principal limitation of the TeMPeST device is the amplitude of the imposed displacements on the surface of the tissue, thus restricting the analysis to the linear domain.

The Creep Indenter (Kerdok, 2006) is a bench-top and open surgical device which observes the large deformation time domain response of tissues. It performs normal indentation tests with large strains typical of surgical manipulations (Figure 9). The device was rigidly mounted to the same platen on which the organ rests to avoid relative motion artifacts. Loads up to 100 g can be applied, generating nominal surface stresses up to 34.7 kPa with a maximum depth of indentation on the order of 12 mm.

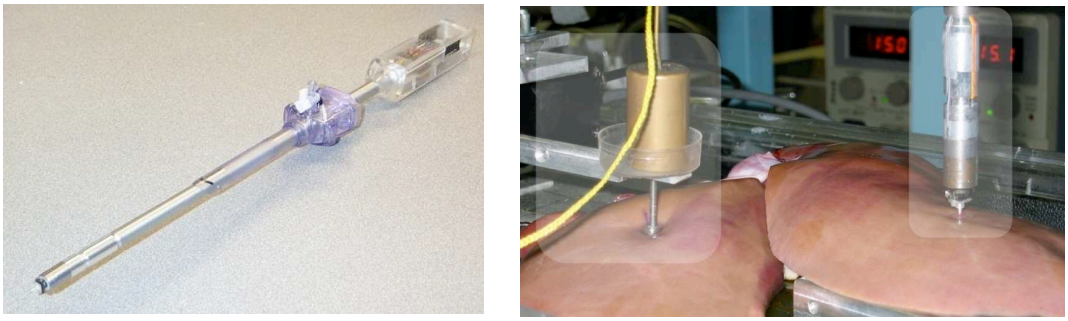


***Figure 9**: (left) TeMPeST 1-D minimally invasive tissue compliance testing device. (right) Creep Indenter on the left and TeMPeST 1-D on the right, measuring perfused porcine liver.*

To compensate for the drawback of the previous measurement devices (i.e. allow large displacements to be applied to the tissue while maintaining *in vivo* characteristics), a new device was developed. The *normo-thermic, extracorporeal perfusion system* (Figure 10) maintains hydration, osmotic and oncotic balance, physiological temperature, and most importantly mean hydrostatic pressure corresponding with normal blood pressure within the tissue. This system is a significant advance over simpler methods of preparing tissues for mechanical testing, and we have shown that this system supports a harvested organ in near *in vivo* biomechanical conditions for over six hours (Kerdok *et al.*, 2005). This allows a much longer testing period than what is possible *in vivo*, (usually a series of experiments of less than a minute) and in a much more controlled environment that can be achieved intra-operatively.

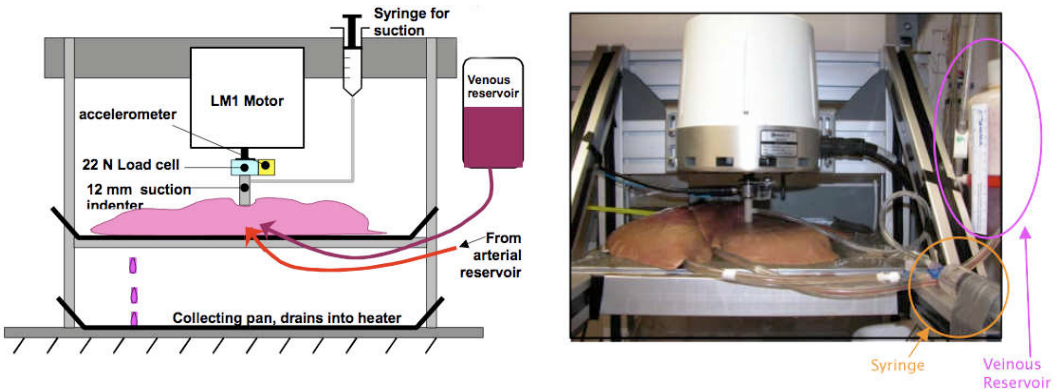


***Figure 10**: (Left) Schematic of the Normo-thermic Extracorporeal Liver Perfusion (NELP) system, and (right) the NELP system in action, testing porcine liver.*

To preserve the organ's mechanical viability during *ex vivo* mechanical testing, the organ's vascular system is flushed with cold veterinary lactated ringer's solution for transport to the mechanical testing laboratory. The *normo-thermic extracorporeal perfusion system* is then used to maintain the mechanical integrity of the whole organ while mimicking physiologic conditions in an *ex vivo* setting. This is achieved by maintaining temperature, hydration, and pressure during indentation testing. For testing the liver, a lactated ringer's solution was pumped to two elevated reservoirs (see Figure 10) in order to hydrostatically maintain physiologic pressures of about 97 mmHg to the hepatic artery and about 18 mmHg to the portal vein. With this setup in place, experiments to characterize the liver and determine a constitutive law can be carried out.

A constitutive model for the liver

In this section is described, through an example, what should be referred to as "model identification". Model identification is different from "parameter estimation"; it is an exercise in assimilating data as well as physical, mechanical, and biological principles into a coherent mechanistic understanding of physiology. On the other hand, parameter estimation involves using observed data to estimate values of parameters based on statistical procedures. To characterize the behavior of the liver observed during the experiments using the *normo-thermic extracorporeal perfusion system* described above, a physiologically derived constitutive model was proposed. Such an approach is particularly relevant in the case of the liver, which behavior under loading conditions can partially be explained by its anatomy and physiology.

Anatomy of the liver

The liver is the largest glandular organ of the body. It weighs about 3 lb (1.36 kg). The liver lies on the right side of the abdominal cavity beneath the diaphragm. It is reddish brown in color and is divided into four lobes of unequal size and shape, and surrounded by a capsule of fibrous connective tissue called Glisson's capsule.

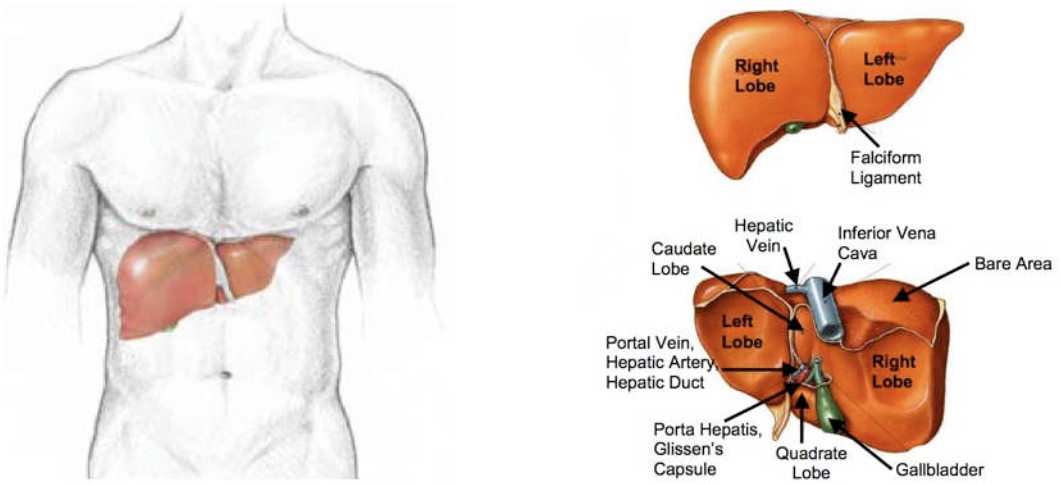


***Figure 11**: (Left): the liver lies on the right side of the abdominal cavity beneath the diaphragm. (Right): it is reddish brown in color and is divided into lobes of unequal size and shape. Blood is carried to the liver via two large vessels called the hepatic artery and the portal vein.*

The parenchyma of the liver is divided into lobules. Blood is carried to the liver via two large vessels called the hepatic artery and the portal vein. The hepatic artery carries oxygen-rich blood from the aorta while the portal vein carries blood containing digested food from the small intestine. These blood vessels subdivide in the liver repeatedly, terminating in very small capillaries. Each capillary leads to a lobule. Liver tissue is composed of thousands of lobules, and each lobule is made up of hepatic cells, the basic metabolic cells of the liver. The liver has many functions, such as breaking down fats, converting glucose to glycogen, producing urea, filtering harmful substances from the blood, etc.

Viscoelastic model

A soft tissue model, also called constitutive law, should describe the properties of its constituents, independent of size or geometry, and specifically how a it deforms in response to applied forces. For most inert materials, measurement of the forces and deformations is straightforward by means of commercially available devices or sensors that can be attached to a test specimen. Many materials, ranging from steel to rubber, have linear constitutive laws, with the proportionality constant (elastic modulus) between the deformation and applied forces. While the same basic principles apply to living tissues, the complex composition of tissues makes obtaining constitutive laws difficult.

From a physiological and biomechanical standpoint, the liver can be described as being mainly comprised of a tough collagenous capsule, vast cellular parenchyma, and pressurized fluid-filled vessels. Each of these major constituents has an impact on the overall behavior of the liver, and the choice of a constitutive law. Consequently, we have modeled the liver as a homogenous, initially isotropic, fluid-filled structure, containing a hyperelastic collagen network in series with a viscoelastic cellular network. A third porous network represents the volumetric response due to the fluid flow. Finally, the tough outer capsule structure is modeled as a hyperelastic collagenous membrane. The nonlinear, viscoelastic, strain-rate dependent response to large deformations was found to be sensitive to its geometric and physiologic boundary conditions. More details about the analysis of the data and the proposed model can be found in Amy Kerdok's PhD thesis (Kerdok, 2006).

The proposed model is a viscoelastic modification of the Arruda-Boyce hyperelastic model whose global response combines contributions from the tissues microstructure in the form of three parallel networks (Arruda *et al.*, 1993). The strain energy potential U for the Arruda-Boyce model is described as

$$U = \mu \sum_{i=1}^{5} \frac{C_i}{\lambda_m^{2i-2}} \left(\overline{I}_1^i - 3^i \right) + \frac{1}{D} \left(\frac{J^2 - 1}{2} - \ln J \right) \quad (2)$$

with $C_1 = \frac{1}{2}$, $C_2 = \frac{1}{20}$, $C_3 = \frac{11}{1050}$, $C_4 = \frac{19}{7000}$, $C_5 = \frac{519}{673750}$

where $\overline{I}_1 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2$ is the first deviatoric strain invariant, and λ_i the principle stretches; λ_m is the limiting chain extensibility, μ the initial shear modulus, J the elastic volume ratio, and D is a temperature dependent material parameter related to the bulk modulus.

The three parallel networks of the liver constitutive model include a hyperelastic 8-chain network (Eq. 2) to capture the elastic response from collagen and elastin fibrils, a viscoelastic network to

establish the time dependent response from the parenchymal cells and the extracellular fluid, and a network that provides an induced pressure from the vascular flow within the organ using a linear Darcy's law (fluid flow through a porous medium). Details can be found in (Kerdok, 2006). The complete model is characterized by twelve parameters, of which seven are independent of each other. The first two networks characterize the isochoric response of the tissue (viscohyperelastic deformation at constant volume), while the last one describes the volumetric transient response (see Figure 12). We have created an algorithm for solving for the seven parameters through the use of inverse finite element modeling. This model is suitable for broader application to other tissues, having been developed in the context of human cervix tissue modeling, and applied to breast carcinoma tissue samples.

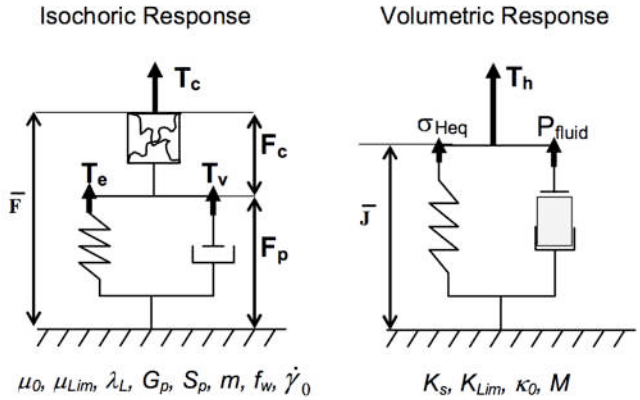


***Figure 12**: Rheological representation of constitutive model separated into the isochoric (deviatoric) response and volumetric (hydrostatic) response.*

Parameter Estimation

Solving the inverse problem (identifying the material parameters from the known force-displacement response) requires replication of the experimental boundary conditions in the form of a finite element model (Figure 13). Either the force or the displacement data are used as prescribed inputs to the model. The constitutive relation and its material parameters govern the model's response to the prescribed loading condition. Initial estimates are given to the material parameters and an error estimate between the model's response and the experimental data is determined. The parameters are then updated in an iterative fashion until the model's result matches that of the data. To solve the inverse problem, the model was implemented using a commercial finite element software. Given the length scale of the liver lobe (~250 x 130 mm) to the size of the indenter (6-12 mm diameter), the indentation tests were represented with an axisymmetric geometry assuming an infinite half-space (Figure 13). The width (*w*) and height (*h*) of the mesh were changed according to the actual dimensions of the liver.

The tissue was modeled as a homogenous, deformable, solid with 8-node, axisymmetric, thermally-coupled, quadrilateral, biquadratic, temperature elements. The mesh was biased to have increased density in the area under the indenter, but kept as coarse as possible to decrease computation time. An independent model was created to solve for the capsule parameters from the data obtained through uniaxial tension experiments.

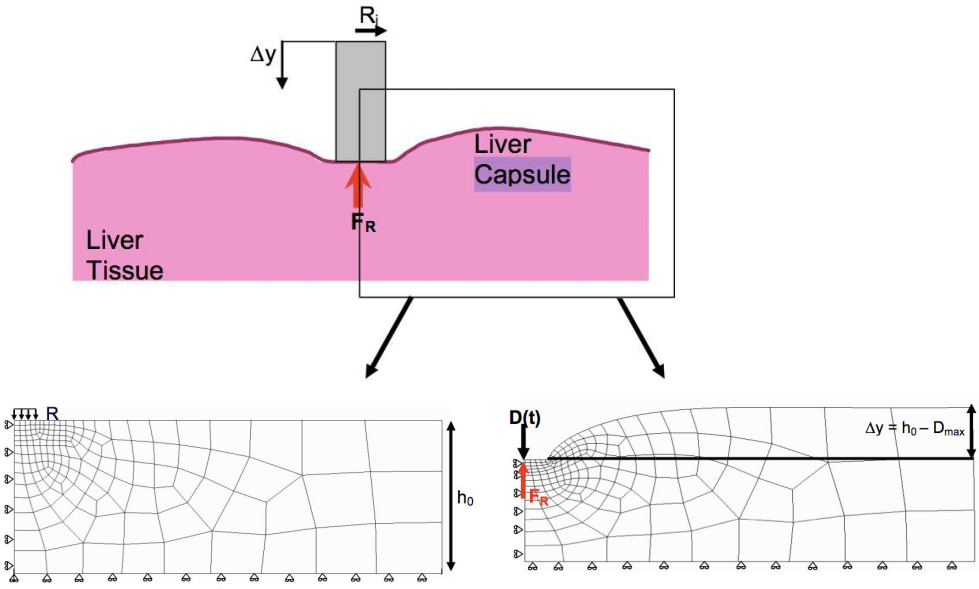


***Figure 13**: (Top): a schematic representation of the indentation experimental conditions. (Bottom left): the axisymmetric meshed geometry and boundary conditions used in ABAQUS to represent the experimental conditions. (Bottom right): an example of a deformed mesh showing the reaction force on the indented area.*

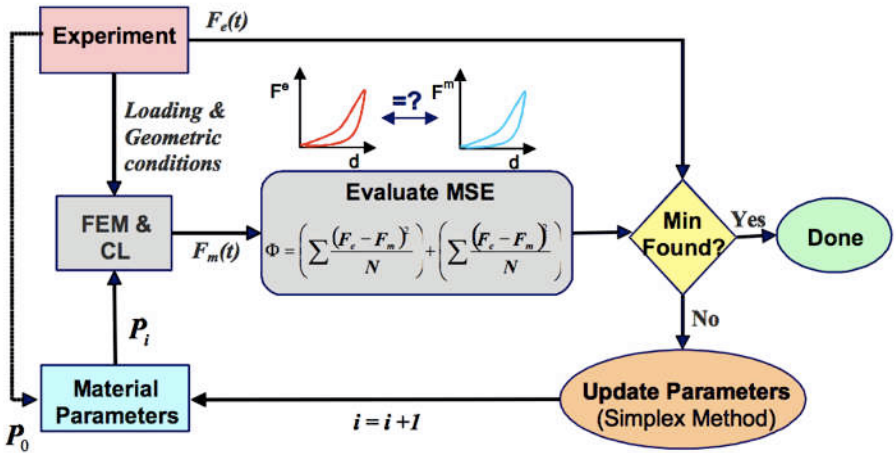


***Figure 14**: diagram illustrating the iterative algorithm used to solve the inverse problem with a finite element model.*

The parameter space of the constitutive model was systematically explored and the results compared to the data using an optimization scheme (Figure 14). The objective function Φ to be minimized was defined using the mean square error between the data and the model of the force versus displacement (or time in the case of multiple indentations) curve and the time derivative of that curve

$$\Phi = \left(\sum \frac{\left(F_{\text{data}} - F_{\text{model}}\right)^2}{n} \right) + \left(\sum \frac{\left(\dot{F}_{\text{data}} - \dot{F}_{\text{model}}\right)^2}{n} \right) \quad (3)$$

An iterative adjustment of the material parameters was then performed according to the Nedler-Mead simplex method (Lagarias *et al.*, 1998). The simplex was allowed to only sample the solution space within the bounds given for each parameter. After determining a set of ”final” parameter values for the liver, the model response was compared to actual experimental data. The computed mean squared error between the model and experimental data was found to be relatively small (MSE = 0.17) thus showing a good predictive capability for the model.

Interactive soft tissue manipulation

Visual realism and real-time interactions are essential in medical simulation: real-time interaction requires that any action from the operator generates an instantaneous response from the simulator, regardless of the complexity of the simulated anatomy. Since the majority of medical procedures involve non-rigid anatomical structures, their shape needs to be recomputed during a simulated procedure. If only visual feedback is required, this information needs to be computed at about 25Hz. When force feedback devices are used in the simulation system, precise computation of the interaction forces is also required, and the rate of the computation is usually in the hundreds of Hertz. Achieving fast computation of deformation and related forces, for complex geometrical models, is a major challenge for medical simulation.

Computing soft-tissue deformation in real-time

Once tissue properties and constitutive laws are known for a given anatomical structure, algorithms for real-time computation of the tissue deformation must be implemented. Such algorithms have been developed during the past ten years, taking their roots both in Computer Graphics and Continuum Mechanics. Terzopoulos (Terzopoulos, *et al.* 1987), Waters (Waters *et al.*, 1992) and Platt (Platt *et al.*, 1988) showed the advantages of physically-based models over traditional models used in computer animation. However, it is important to separate models derived from the laws of continuum mechanics or biomechanics from models that, even if they follow the laws of physics, are more ad-hoc representations. For instance, early works on surgery simulation used rather simple models to provide real-time interactions, such as mass-spring systems (Kühnapfel *et al.*, 1995). This type of representation of a deformable body has the benefit to be simple to implement and computationally efficient. There are, however, many drawback to this initial representation which have led to several improvements, in terms of modeling capabilities and in terms of computational efficiency (Provot, 1995) (Sørensen *et al.*, 2006). This modeling approach might best be described as “quasi-physical”: although the underlying mathematics has a basis in classical physics, it does not represent material properties directly. As a consequence, this type of approach tends to be replaced by models derived from continuum mechanics, particularly models based on the laws of elasticity, solved using finite element techniques. However, various modifications and simplifications need to be introduced to achieve real-time or interactive simulations.

In the remainder of this section, I will make an attempt at covering most of the work that was done in (or is relevant to) the field of medical simulation. For this, I will take a model based point of view, since different physical models can be applied to similar applications, i.e. we can compute the deformation of an elastic solid using various underlying models, such as mass-spring systems, finite elements or mesh-free methods. I will also treat the problem of time integration (or time discretization) separately since it is inherent to all dynamic simulations.

Continuum Elasticity

Here we describe a Lagrangian view of the deformation of an object, which is the most common approach, while Eulerian descriptions are more typically used for modeling fluids. A deformable object is typically defined by its undeformed shape (also called equilibrium configuration, rest shape or initial shape) and by a set of material parameters that define how it deforms under external forces (loads). If we define the rest shape as a connected subset M of $\mathbb{R}^3$ then the coordinates $\mathbf{m} \in M$ of a point in the object are called material coordinates of that point. In the discrete case M is a discrete set of points that sample the rest shape of the object (see Figure 15). When external forces are applied, the object deforms and a point originally at location $\mathbf{m}$ (i.e. with material coordinates m) will move to a new location $\mathbf{x}(\mathbf{m})$, where $\mathbf{x}(\mathbf{m})$ defines the spatial coordinates of that point. Since new locations can be defined for all material coordinates $\mathbf{m}$, $\mathbf{x}$ is a vector field defined on M. Alternatively, the deformation can also be specified by the displacement vector field $\mathbf{u}(\mathbf{m}) = \mathbf{x}(\mathbf{m}) - \mathbf{m}$ defined on M.

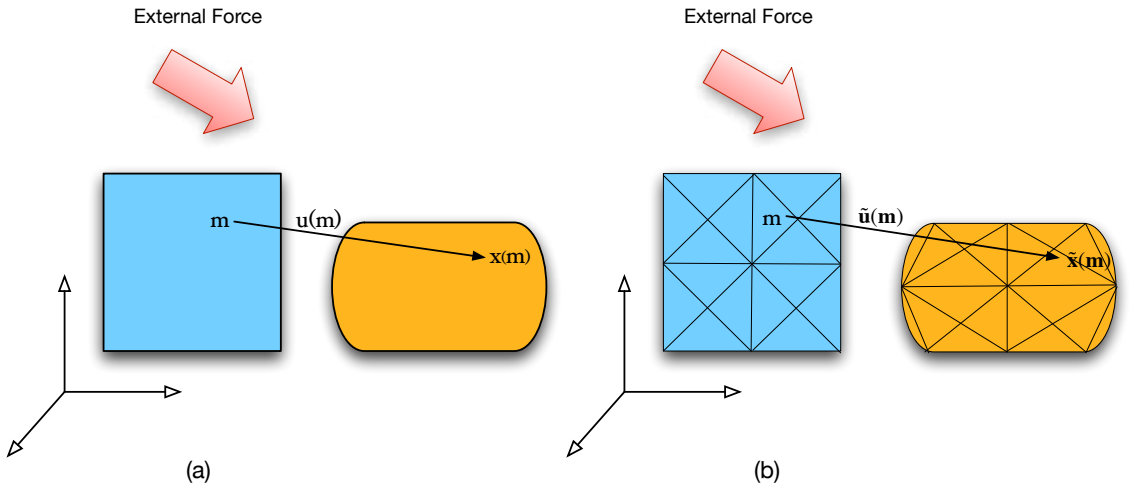


***Figure 15**: In the Finite Element method, a continuous deformation (a) is approximated by a sum of (linear) basis functions defined inside a set of finite elements (b).*

From $\mathbf{u}(\mathbf{m})$ the elastic strain ε can be computed (ε is a dimensionless quantity which, in the linear one-dimensional case, is simply $\Delta l/l$). A spatially constant displacement field represents a translation of the object with no strain. Therefore, it becomes clear that strain must be measured in terms of spatial variations of the displacement field $\mathbf{u} = \mathbf{u}(\mathbf{m}) = (u,v,w)^{\mathrm{T}}$. Two frequently used elastic strains in the field of real-time simulation are

$$\varepsilon_G = \frac{1}{2}\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T + (\nabla\mathbf{u})^T \nabla\mathbf{u}\right) \quad (4)$$

and $$\varepsilon_C = \frac{1}{2}\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right) \quad (5)$$

where the symmetric tensor $\varepsilon_G \in \mathbb{R}^{3x3}$ is Green's nonlinear strain tensor and $\varepsilon_C \in \mathbb{R}^{3x3}$ its linearization, called Cauchy's linear strain tensor. The gradient of the displacement field is the 3 by 3 matrix below

$$\nabla \mathbf{u} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} & \frac{\partial u}{\partial z} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} & \frac{\partial v}{\partial z} \\ \frac{\partial w}{\partial x} & \frac{\partial w}{\partial y} & \frac{\partial w}{\partial z} \end{bmatrix} \quad (6)$$

The material law (or constitutive law) is used for the computation of the symmetric internal stress tensor $\sigma \in \mathbb{R}^{3x3}$ for each material point m based on the strain ε at that point (σ is measured as force per unit area, in Pascal with 1 Pa = 1 N/m^2). Most publications in the field of interactive (medical) simulation rely on Hooke's linear material law $\sigma = E\ \varepsilon$ where E is a rank four tensor which relates the coefficients of the stress tensor linearly to the coefficients of the strain tensor. For isotropic materials, the coefficients of E only depend on Young's modulus and Poisson's ratio.

Finite Element Method

The Finite Element Method (FEM) is a popular method for finding approximate solutions of Partial Differential Equations (PDE's) on irregular grids or meshes. The finite-element method can be traced back to the middle of the 20th century and originated from the needs for solving complex structural analysis problems in civil engineering and aeronautical engineering. Although various initial approaches were introduced, they all shared one essential characteristic: mesh discretization of a continuous domain into a set of discrete sub-domains. In our case, the volume of the deformable object is generally discretized using a mesh composed of linear elements (triangles, tetrahedra, hexahedra, etc.) where shape functions with limited support are defined. This leads to continuous representations with varying levels of continuity (linear, quadratic, etc.). In general the FEM approach requires considerable computation, which is not a critical issue for many engineering applications, where accuracy of the solution is the prime concern. Yet, this major limitation is balanced by an essential feature of FEM approaches: in comparison with mass-spring systems, the FEM method gives a much clearer representation of tissue parameters to assess forces and damage and the accuracy of the model can much more readily be upgraded by increasing the number of elements. However, it is essential to keep in mind that the FEM is only a numerical technique, and that it is essentially the choice of the constitutive model (and its related PDEs) that will dictate the behavior of the deformable model. In the field of medical simulation, most applications are based on elasticity theory, but the FEM technique can be used for other simulations, such as physiology. We will describe such an example in Section IV of this chapter with the computation of arterial laminar flow.

Mathematical basis

The partial differential equation governing dynamic elastic materials is given by

$$\rho \ddot{\mathbf{x}} = \nabla \cdot \sigma + \mathbf{f} \quad (7)$$

where ρ is the mass density of the material and $\mathbf{f}$ are externally applied forces such as gravity or collision forces. The divergence operator turns the 3x3 stress tensor back into a 3-vector

$$\nabla \cdot \sigma = \begin{bmatrix} \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z} \\ \frac{\partial \sigma_{yx}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yz}}{\partial z} \\ \frac{\partial \sigma_{zx}}{\partial x} + \frac{\partial \sigma_{zy}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} \end{bmatrix} \quad (8)$$

representing the internal force resulting from a deformed infinitesimal volume. Equation (Eq. 7) shows the equation of motion in differential form in contrast to the integral form which is used in the Finite Volume method. The Finite Element Method is used to turn a PDE into a set of algebraic equations which are then solved numerically. To this end, the domain M is discretized into a finite number of disjoint elements (i.e. a mesh). Instead of solving for the spatially continuous function $\mathbf{x}(\mathbf{m}, t)$, one only solves for the discrete set of unknown positions $\mathbf{x}_i(t)$ of the nodes of the mesh. First, the function $\mathbf{x}(\mathbf{m}, t)$ is approximated using the nodal values by

$$\tilde{\mathbf{x}}(\mathbf{m},t) = \sum_i \mathbf{x}_i(t)\mathbf{b}_i(\mathbf{m}) \quad (9)$$

where $\mathbf{b}_i()$ are fixed nodal basis functions which value is 1 at node i and zero at all other nodes, also known as the Kronecker Delta property. In the most general case of the Finite Element Method, the basis functions do not have this property. In that case, the unknowns are general parameters which can not be interpreted as nodal values. Substituting $\tilde{\mathbf{x}}(\mathbf{m},t)$ into Eq. 8 results in algebraic equations for the $\mathbf{x}_i(t)$. In the Galerkin approach, finding the unknowns $\mathbf{x}_i(t)$ is viewed as an optimization process. When substituting $\mathbf{x}(\mathbf{m},t)$ by the approximation $\tilde{\mathbf{x}}(\mathbf{m},t)$ the infinitely dimensional search space of possible solutions is reduced to a finite dimensional subspace. In general, no function in that subspace can solve the original PDE. The approximation will generate a deviation or residue when substituted into the PDE. In the Galerkin method, the approximation which minimizes the residue is sought, i.e. an approximation whose residue is perpendicular to the subspace of functions defined by (Eq. 9) is computed.

Very often, a simple form of the Finite Element method is used for the interactive simulation of deformable objects. In this method, both the masses and the (internal and external) forces are lumped to the vertices. The nodes in the mesh are treated like mass points in a mass-spring system while each element acts like a generalized spring connecting all adjacent mass points. The forces acting on the nodes of an element due to its deformation are computed as follows (see for instance (O'Brien et al., 1999): given the positions of the vertices of an element and the fixed basis functions, the continuous deformation field $\mathbf{u}(\mathbf{m})$ inside the element can be computed using (Eq. 10). From $\mathbf{u}(\mathbf{m})$, the strain field $\varepsilon(\mathbf{m})$ and stress field $\sigma(\mathbf{m})$ are computed. The deformation energy of the element is then given by

$$E = \int_V \varepsilon(\mathbf{m}) \bullet \sigma(\mathbf{m}) \; d\mathbf{m} \quad (10)$$

where the dot represents the component-wise scalar product of the two tensors. The forces can then be computed as the derivatives of the energy with respect to the nodal positions. In general, the relationship between nodal forces and nodal positions is nonlinear. When linearized, the relationship for an element e connecting n_e nodes can simply be expressed as

$$\mathbf{f}_e = \mathbf{K}_e \mathbf{u}_e \quad (11)$$

where $\mathbf{f}_e \in \mathbb{R}^{3n_e}$ contains the n_e nodal forces and $\mathbf{u}_e \in \mathbb{R}^{3n_e}$ the n_e nodal displacements of an element. The matrix $\mathbf{K}_e \in \mathbb{R}^{3n_e \times 3n_e}$ is called the stiffness matrix of the element. Because elastic forces coming from adjacent elements add up at a node, a stiffness matrix $\mathbf{K} \in \mathbb{R}^{3n \times 3n}$ for an entire mesh with n nodes can be formed by assembling the element's stiffness matrices

$$\mathbf{K} = \sum_e \mathbf{K}_e \quad (12)$$

In the general case, if we let $\mathbf{u} = \mathbf{x} - \mathbf{x_0}$, the equation of motion in the dynamic case can be written

$$\mathbf{M}\ddot{\mathbf{u}} + F(\dot{\mathbf{u}}, \mathbf{u}) = \mathbf{f}(\mathbf{u}) \quad (13)$$

where $\mathbf{f}(\mathbf{u})$ describes non-linear external forces and $F(\dot{\mathbf{u}}, \mathbf{u})$ is a non-linear function that can be approximated by $F(\dot{\mathbf{u}}, \mathbf{u}) = F(\dot{\mathbf{u}} + d\dot{\mathbf{u}}, \mathbf{u} + d\mathbf{u}) - \mathbf{D}d\dot{\mathbf{u}} - \mathbf{K}d\mathbf{u}$. In the case of small displacements, it becomes $F(\dot{\mathbf{u}}, \mathbf{u}) = \mathbf{D}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u}$ thus leading to the more common equation of motion in the linear case

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f} \quad (14)$$

where $\mathbf{M} \in \mathbb{R}^{n \times n}$ is the mass matrix, $\mathbf{D} \in \mathbb{R}^{n \times n}$ the damping matrix and $\mathbf{f} \in \mathbb{R}^n$ externally applied forces. If we consider that mass and damping effects are concentrated at the nodes (technique called mass lumping) then $\mathbf{M}$ and $\mathbf{D}$ are diagonal matrices. In this case, $\mathbf{M}$ just contains the point masses of the nodes of the mesh on its diagonal. The damping matrix $\mathbf{D}$ is often defined as a linear combination of the stiffness matrix $\mathbf{K}$ and the mass matrix $\mathbf{M}$, and is referred to as Raleigh damping

$$\mathbf{D} = \alpha\mathbf{M} + \beta\mathbf{K} \quad (15)$$

A vast majority of the work done in the area of real-time deformable models using the Finite Element Method was based on linear elasticity equations, see (Bro-Nielsen *et al.*, 1996) (Cotin *et al.*, 1999) (Picinbono *et al.*, 2001) (Debunne *et al.*, 2002) for instance. Linear partial differential equations yield linear algebraic systems which can be solved more efficiently and more stably than nonlinear ones. Unfortunately, linearized elastic forces are only valid for small deformations, in particular large rotational deformations lead to highly inaccurate deformations. This problem, and the general case of geometric non-linearities, can be solved using different techniques. A recent and popular approach among the computer graphics and medical simulation communities is the co-rotational method (Felippa, 2000) (Muellller *et al.*, 2004). Using this approach, the deformation of an element e is decomposed into a rigid transformation component and a "pure" deformation component. This relies essentially on the computation of the rotation of the element from its rest configuration. This can be achieved in different ways, such as a polar decomposition (Muellller *et al.*, 2004) or through a geometrical analysis (Nesme *et al.*, 2005). The linear equation (Eq. 11) for the elastic forces of an element is then replaced by

$$\mathbf{f}_e = \mathbf{R}_e \mathbf{K}_e \mathbf{R}_e^T (\mathbf{x} - \mathbf{x}_0) \quad (16)$$

where $\mathbf{R}$ is a matrix that contains n 3x3 identical rotation matrices along its diagonal (n is the number of nodes in the element). The vector $\mathbf{x}$ contains the actual positions of the nodes of the element while $\mathbf{x_0}$ contains their rest positions.

Such a technique overcomes some of the limitations of linearly elastic models, without compromising too much the computational efficiency. However, as long as the equation of motion is integrated explicitly in time, nonlinear elastic forces resulting from Green's strain tensor pose no computational problems. The nonlinear formulas for the forces can simply be evaluated and used directly to integrate velocities and positions as in (O'Brien *et al.*, 1999) (Picinbono *et al.*, 2001).

Real-time approaches

While the Finite Element Method is an unavoidable choice in many engineering disciplines, its use in the context of real-time simulation is far from being straightforward, as this method is known to be highly computationally expensive. The first attempts to achieve interactive simulations using Finite Element Methods were limited to very coarse meshes and only used surface representations, which do not characterize the volumetric nature of the vast majority of anatomical structures. Our main contribution in this area consists of three pioneering articles, two of which were co-authored with Morten Bro-Nielsen. These articles, published in 1996 and 1999 present two different approaches to the problem of real-time or interactive simulation using volumetric FEM representations.

The first contribution (Bro-Nielsen *et al.*, 1996) was based on the concept of condensation, a technique that allows a significant speedup by simulating only the visible surface nodes of a volumetric mesh. This paper also introduced two additional improvements, aimed at speeding up the computation of the solution through optimized matrix-vector operations. The condensation technique allows to compress the linear matrix system resulting from a volumetric finite element analysis of a linear elastic problem to a system with the same complexity as a surface finite element representation of the same object. The second improvement concerns the way the resulting linear system is used in the simulation. In contrast with traditional approaches, we explicitly invert the system and use matrix-vector multiplications with this matrix to achieve low computation times. The third improvement is called Selective Matrix-Vector Multiplication and takes advantage of the sparse nature of the vector of external forces. If we consider the linear matrix system $\mathbf{Ku} = \mathbf{f}$ corresponding to the static case of equation (Eq. 14) the vector of displacements $\mathbf{u}$ contains both nodes located on the surface and within the volume of the mesh. In most simulations, we are only interested with the behavior of surface nodes, since they are the only visible nodes. Hence the idea of using a condensation technique to "remove" the internal nodes from the system.

Without loss of generality, let us assume that the nodes of the mesh are ordered with the surface nodes first. We can then rewrite the linear system $\mathbf{Ku} = \mathbf{f}$ as a block matrix system

$$\begin{bmatrix} \mathbf{K}_{ss} & \mathbf{K}_{si} \\ \mathbf{K}_{is} & \mathbf{K}_{ii} \end{bmatrix} \begin{bmatrix} \mathbf{u}_s \\ \mathbf{u}_i \end{bmatrix} = \begin{bmatrix} \mathbf{f}_s \\ \mathbf{f}_i \end{bmatrix} \quad (17)$$

From this block matrix system, we can define a new linear matrix system $\mathbf{K}_s^* \mathbf{u}_s = \mathbf{f}_s^*$ which only involves surface nodes

$$\begin{cases} \mathbf{K}_s^* = \mathbf{K}_{ss} - \mathbf{K}_{si}\mathbf{K}_{ii}^{-1}\mathbf{K}_{is} \\ \mathbf{f}_s^* = \mathbf{f}_s - \mathbf{K}_{si}\mathbf{K}_{ii}^{-1}\mathbf{f}_i \end{cases} \quad (18)$$

Generally, the new stiffness matrix is dense compared to the sparse structure of the original. Consequently, using a pre-computation stage when the matrix is inverted once for the whole simulation is relevant here. The main issue in performing explicit matrix inversion (rather than solving the linear system using a Conjugate Gradient method for instance) lies in the large amount of memory required to store the inverted matrix (which is dense). Matrix operations need also to be performed with care to reduce numerical errors. Yet, with recent computers, memory is less of an issue, and the meshes we manipulate in medical simulation are less refined than they are usually in mechanical engineering applications. Once the inverse of condensed stiffness matrix has been computed, the computation of the displacement vector is very fast, due to the reduced number of degrees of freedom. Comparing this computation (without including the time necessary for the pre-computation of the inverse) against various other techniques (including pre-conditioned conjugate gradient methods) showed that our approach was at least 10 times faster.

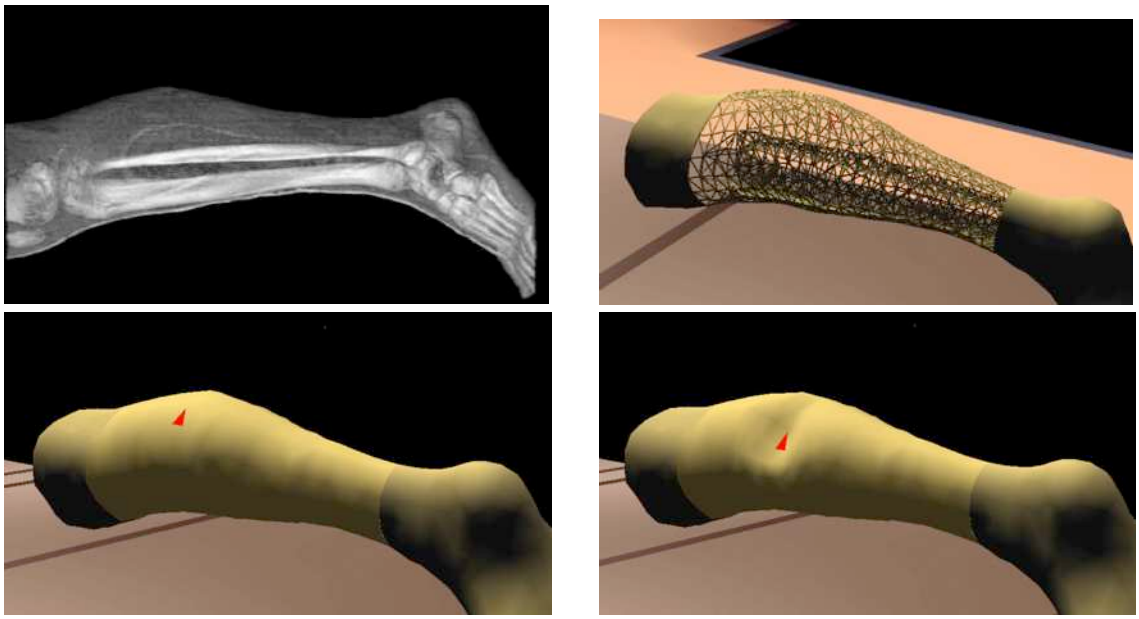


***Figure 16**: (Top Left): Voxel data of the leg from the Visible Human data set, (Top Right): Wireframe model of the lower leg, (Bottom Left): simulation of palpation on the lower leg, the point of application if the red triangle, and (Bottom Right): the resulting deformation computed in real-time.*

To improve further the speed up of the method, we proposed an optimization called the Selective Matrix Vector Multiplication (SMVM). This applied in particular in simulations where external forces apply to only a few nodes (such as the example of Figure 16). In that case, only a few elements of the force vector are non-zero and standard matrix vector multiplication would involve a large number of redundant multiplications. We note that

$$\mathbf{u} = \mathbf{K}^{-1}\mathbf{f} = \sum_i \mathbf{K}_{*i}^{-1}\mathbf{f}_i \quad (19)$$

where $\mathbf{K}^{-1}_{*i}$ is the i^{th} column vector of $\mathbf{K}^{-1}$ and $\mathbf{f_i}$ is the i^{th} element of $\mathbf{f}$. Since the majority of the $\mathbf{f_i}$ are non-zero, we restrict i to run through only the positions of $\mathbf{f}$ for which $\mathbf{f_i} \neq 0$. If n of the N elements of $\mathbf{f}$ are non-zero, this will reduce the complexity to $O(n/N)$ times the time of a normal matrix vector multiplication.
The techniques presented above were applied to the simulation of lower leg surgery (Figure 16), as part of a bigger program aimed at open surgery simulation. The simulation ran at about 20 Hz on a SGI, and used different threads for the computation and the visualization of the virtual operating field.

The second contribution (Cotin *et al.*, 1996a) and (Cotin *et al.*, 1999) was based on an extension of the superposition principle, which allows to describe the vector of external forces $\mathbf{f}$ as a linear combination of independent external forces. For such a principle to apply, we limited our model to the linear domain, but proposed several key ideas to enable very fast computation of the deformation, and allowed haptic feedback to be used in the simulation. We applied this approach to a training system for laparoscopic surgery (specifically hepatic surgery).

Starting from (Eq. 14) we made the assumption that the model could be considered static, as the velocity of the surgical instruments is reduced in such procedures. We also considered the transient dynamic response of the deformable organ to be negligible. Also, to be able to use a force feedback device in the simulation, we proposed to model the interactions using constraints rather than external forces. In a general approach, external forces are applied to the surface of the deformable solid, as illustrated in (Figure 16). However, when using a force feedback device for the interactions with the deformable model it is usually not possible to measure the forces exerted by the operator. The position of the end effector of the laparoscopic device is the only information transmitted by the haptic interface. Consequently, the deformation must be driven by specific constraints in displacement and not in force. Hence our boundary conditions are mainly the contact points displacements between the surgical tool and the body. We can then deduce both the forces exerted on the end effector of the tool and the global deformation by applying the linear elasticity equations. The computed forces are finally set into the force feedback system in order to provide a mechanical resistance into the surgeon's hands: the loop is closed. In order to include the new constraints, we replace the linear system $\mathbf{Ku} = \mathbf{f}$ by the following extended system

$$\begin{bmatrix} \mathbf{K} & \bar{\mathbf{K}} \\ \bar{\mathbf{K}}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{u}^* \end{bmatrix} \quad (20)$$

where $\bar{\mathbf{K}}$ is a matrix composed of vectors $\mathbf{e}_i$ with a 1 in the i^{th} position and zeros elsewhere. This system results from the Lagrange multipliers method which is used for the imposition of specified values for the solution variables. The values λ_i of $\boldsymbol{\lambda}$ obtained after solving the system (Eq. 20) are equal to the opposite of the force that needs to be applied at the degree of freedom u_i in order to impose the displacement $u_i = u_i^*$.

In linear theory, the behavior of the deformable model is physically correct only for small displacements (about 5% of the mesh size), and it is less realistic for larger deformations, as well for large rotations. Although it is a major disadvantage of linear elasticity, the integration of force feedback in the simulation can help limit the range of deformations applied to the deformable model (assuming it is not too soft). As the deformation increases, the force in the user's hand will increase, thus preventing large deformations to occur. Under this assumption of linear elastic behavior, we can use an extension of the superposition

principle to compute any mesh deformation from the knowledge of a finite set of elementary deformations. The method relies on two separate stages: a pre-computation stage, done off-line, which computes the deformation of the whole mesh from elementary constraints, and the use of this pre-computed information to perform real-time deformation and haptic feedback. The pre-processing algorithm can be described as follows:

- Specify a set of mesh nodes that remain fixed during the deformation. Specified values $u_i^* = 0$ are not necessarily set for all three degrees of freedom of the nodes.
- For each degree of freedom ki of each "free" node k on the surface of the mesh, an elementary displacement constraint u_i^* is set and the following quantities are computed:
  - the displacement of every free node n ($n \neq k$) in the mesh. This displacement is stored as a set of 3x3 tensors $\mathbf{T}_{nk}^u$ expressing the relation between the displacement of node n and the elementary displacement u_k^* imposed at node k. The applied displacement is the same for every node.
  - the components of the elementary force -$\boldsymbol{\lambda}$ at node k. It is stored as a 3x3 tensor $\mathbf{T}_k^f$.

If m is the number of free nodes, the linear system

$$\begin{bmatrix} \mathbf{K} & \bar{\mathbf{K}} \\ \bar{\mathbf{K}}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{u}_k^* \end{bmatrix} \quad (21)$$

has to be solved $3m$ times. We use an iterative method (conjugate gradient) to solve each linear system (Eq. 20). The pre-processing stage can last anywhere from a few minutes to several hours depending on the size of the model and on the desired accuracy. Once computed, the result of the pre-processing stage can be stored in a file and used for any simulation, as long as it is based on the same mesh, the same elastic parameters, and initial boundary conditions.

The displacement u_n of a node n in the mesh induced by the constraint u_k^* imposed at node k, can be obtained by the following linear equation

$$u_n = \mathbf{T}_{nk}^u \frac{u_k^*}{\|u_k^*\|} \quad (22)$$

for any node $k \neq n$. In general, more than one node is moved on the surface of the mesh during a contact with another object. The total displacement of a node is the sum of all the displacements induced by the controlled nodes u_{kl}^* with $l=1,..,m$. However, the superposition principle is not directly applicable since the controlled nodes are also influenced by other controlled nodes. For example, considering two nodes with controlled displacements u_{k1}^* and u_{k2}^* we can see on Figure 17 that the final displacement at these nodes will be greater than the imposed displacement. In other words, the mesh would deform more than it should.

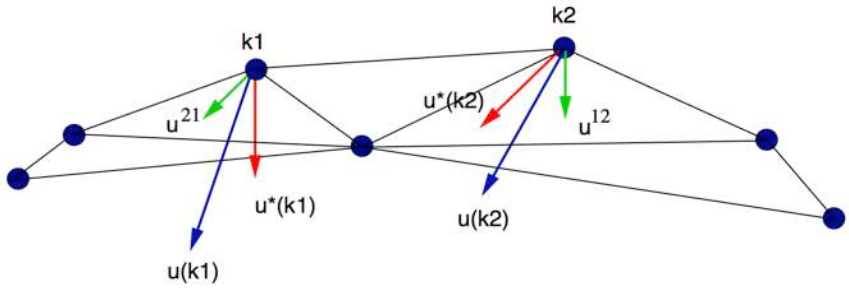


Figure 17: u_{k1}^* and u_{k2}^* are the initial imposed displacements. u^{12} and u^{21} are, respectively, the displacement induced by u_{k1}^* on node k_2 and the displacement induced by u_{k2}^* on node k_1. Finally u_{k1} and u_{k2} are the displacements resulting from the application of the superposition principle.

As a result, the superposition principle needs to be extended to account for the superposition of displacements, not forces. A different set of constraints $\tilde{u}_{kl}^*$ must be applied instead of u_{kl}^* where $\tilde{u}_{kl}^*$ can be determined from the deformation tensors $\mathbf{T}_{nk}^u$. The following matrix represents the mutual influence of all the controlled nodes

$$\mathbf{S} = \begin{bmatrix} \delta & \mathbf{T}_{21}^u & \mathbf{T}_{31}^u & \cdots & \mathbf{T}_{m1}^u \\ \mathbf{T}_{12}^u & \delta & \mathbf{T}_{32}^u & \cdots & \mathbf{T}_{m2}^u \\ \mathbf{T}_{21}^u & \mathbf{T}_{23}^u & \delta & \cdots & \mathbf{T}_{m3}^u \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{T}_{1m}^u & \mathbf{T}_{1m}^u & \mathbf{T}_{1m}^u & \cdots & \delta \end{bmatrix} \quad \text{with} \quad \delta = \begin{bmatrix} u_{k1}^* & 0 & 0 \\ 0 & u_{k2}^* & 0 \\ 0 & 0 & u_{k3}^* \end{bmatrix} \quad (23)$$

The vector of modified constraints $\tilde{\mathbf{u}}^*$ that must be applied to the mesh is determined by

$$\tilde{\mathbf{u}}^* = \mathbf{S}^{-1}\mathbf{u}^* \quad (24)$$

Consequently the extension of the superposition principle can be applied to compute the total and exact displacement of each node n

$$\mathbf{u}_n = \sum_{l=1}^{m} \mathbf{T}_{nkl}^u \tilde{\mathbf{u}}_{kl}^* \quad (25)$$

Finally, the force that should be applied to node k to produce the displacement $\tilde{\mathbf{u}}^*$ is determined by

$$\mathbf{f}_k = \mathbf{T}_{nk}^f \mathbf{u}^* \quad (26)$$

We have implemented and integrated these different computation schemes in a prototype of laparoscopic surgery simulation (Figure 18). The simulation was implemented on a distributed architecture, composed

of a PC (Pentium processor at 166 Mhz) and a Dec Alpha workstation (with a CPU running at 400 Mhz and a 3D graphics card). The force feedback device was connected to the PC for technical reasons. Thanks to our approach, and as we have seen in equations (Eq. 25) and (Eq. 26), it is possible to split the computation of the forces and the computation of the deformation. The deformation is performed on the Alpha station along with the visual feedback, while the computation of the forces is performed on the PC, which is directly connected to the haptic interface. The communication between the two machines is performed via an ethernet connection. Results are very convincing, since the computation of the deformation of a liver model (containing about 1,500 nodes) could run at 50 Hz, and the computation of the forces could run at a much higher frequency (about 300 Hz) as it involves less computation.

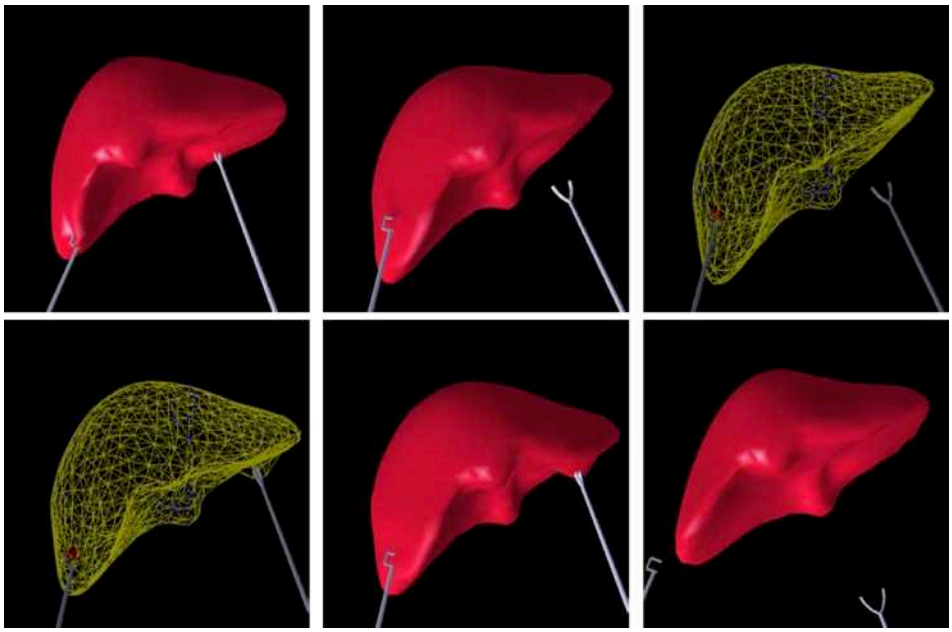


***Figure 18**: Sequence of images issued from a real-time simulation. The liver model contains about 1,500 nodes. The simulation was performed at 50Hz for the visual feedback and 300Hz for force feedback on a Pentium 166 MHz.*

Both the condensation method we proposed (Bro-Nielsen *et al.*, 1996) and banded-matrix techniques developed by Berkley *et al.* (Berkley *et al.*, 1999) use matrix manipulations to remove the need to compute the influence of internal nodes. This means that real-time solutions are possible at the cost of substantial pre-processing steps. The need for pre-computation can be seen as a limiting factor since topological changes of the mesh (such as might be required to represent tearing, cutting or piercing) would require to go through a new pre-processing stage, incompatible with real-time requirements.

A third contribution was then proposed in (Cotin *et al.*, 1999) as a step towards a real-time FEM method compatible with topological changes. Called the tensor-mass model, it uses a pre-computation step (similar as the one described previously) during which deformation tensors are compiled to define standard displacement fields at each movable node in the mesh. Using a conjugate gradient solver, the tensors and mesh can be updated dynamically, allowing topological changes (such as cutting) for meshes of almost a thousand nodes with updates at 40Hz (on a Pentium 233 MHz PC) in a hepatic surgery simulation. The basic principle relies on the pre-computation of the stiffness matrix $\mathbf{K}^{e}_{jk}$ associated each element (tetrahedron in our case) of the finite element mesh

$$\mathbf{K}_{jk}^{e} = \frac{1}{36V^{e}}(\lambda_e \mathbf{M}_k \mathbf{M}_j^T + \mu_e \mathbf{M}_j \mathbf{M}_k^T + \mu_e \mathbf{M}_j \mathbf{M}_k \mathbf{I}_{3\times 3}) \quad (26)$$

where $\mathbf{I}_{3x3}$ is a 3x3 identity matrix, V^e is the volume of the tetrahedron, λ_e and μ_e are the Lame coefficients, and $\mathbf{M}_j$ is a vector directed along the outer normal vector of triangle opposite to $\mathbf{P}^e{}_i$ (j=1,..,4) one of the vertices of the tetrahedron. The norm of $\mathbf{M}_j$ corresponds to twice the area of the triangle opposite to $\mathbf{P}^e{}_i$. We can see that (Eq. 26) depends only on the material properties λ_e and μ_e of the tissue delimited by the volume of the tetrahedron, and the geometry of the elements at its rest shape.

For a given mass point (or vertex) P_i, the elastic force F_i is therefore the sum of all contributions from all adjacent tetrahedra, and can be expressed as

$$\mathbf{F}_i = \mathbf{K}_{ii}\mathbf{u}_i + \sum_{j\in N(i)} \mathbf{K}_{ij}\mathbf{u}_j \quad (27)$$

where $\mathbf{u}_i = \mathbf{P}_i^0\mathbf{P}_i$ and $\mathbf{u}_j = \mathbf{P}_j^0\mathbf{P}_j$; $\mathbf{K}_{ii}$ is the sum of all tensors $\mathbf{K}_{ii}^e$ adjacent to P_i and $\mathbf{K}_{ij}$ is the sum of all tensors $\mathbf{K}_{ij}^e$ adjacent to edge (i,j). We can see that this expression in (Eq. 27) is very similar to the computation of internal forces in a spring-mass model

$$\mathbf{F}_i = \sum_{j\in N(i)} k_{ij}(\left\|\mathbf{P}_i\mathbf{P}_j\right\| - l_{ij}^0)\frac{\mathbf{P}_i\mathbf{P}_j}{\left\|\mathbf{P}_i\mathbf{P}_j\right\|} \quad (28)$$

It is the similarity between these two expressions that motivated our choice for the name “tensor-mass” model. Such a model presents several advantages. The computation of the internal forces being similar to the one of a spring-mass model, we can use it in a dynamic formulation, and rely on an explicit integration scheme (see below) to reach very fast computation times. Actually, we obtained computations times nearly identical to a spring-mass model. Another benefit of using this formulation combined with an explicit integration scheme is that it is possible to perform simple topological changes without have any computational cost. We illustrated this on a simulation of resection in the context of hepatic surgery (Figure 19).

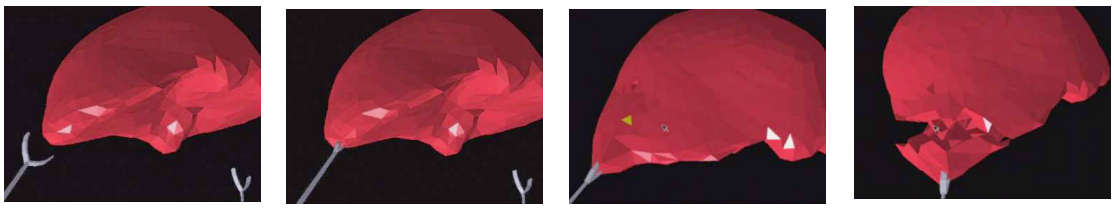


***Figure 19**: Sequence of images extracted from a real-time simulation. The liver model contains about 1,000 nodes. The use of the tensor-mass model allows for simple topological changes (in this case the progressive removal of tetrahedral elements).*

Zhong *et al.* (Zhong *et al.*, 2005) have proposed an extension of our tensor mass model by pre-computing the matrix inverse and condensing to remove fixed boundary nodes. Whereas our method requires that the FEM solver to execute $3k$ times (where k the number of contact nodes), using the inverse allows the

generation of tensor components directly. The authors have also described algorithms to introduce a polynomial interpolation of non-linear behavior and an affine mapping of contact regions using St Venant's Principle. A simulation incorporating these ideas was developed and allowed limited interactions with brain tissue.

The proposed optimized algorithms, and several others such as (James, *et al.*, 1999), rely on a linearization of the constitutive law, thus limiting the field of application of such models. To remedy this issue, other approaches have been proposed. They essentially aim at allowing large displacements (geometric non-linearities) while the stress-strain relationship remains linear (Picinbono *et al.*, 2001). Co-rotational methods, used for instance by (Nesme *et al.*, 2005) or (Mueller *et al.*, 2004) also allow for large displacements or rotations. Techniques based on modal analysis (Basdogan, *et al.*, 2001) (Hauser, *et al.*, 2003) also take advantage of the linear nature of the equations. Finally, multi-scale approaches can accelerate computation time by propagating quickly principal deformations through a coarse mesh and finer deformation information using finer mesh resolutions (Wu *et al.*, 2004) (Debunne *et al.*, 2001). Similar techniques use higher order shape functions rather than a finer mesh (Duriez, et al., 2003) although such approaches have rarely been used in the context of real-time simulation.

Quasi-physical models

Free-form deformations

The earliest attempts to provide interactive simulations using deformable models relied on surface-based representations, the animation of which was achieved by manipulating the nodes of the mesh or outlying control points (Sederberg *et al.*, 1986) (Terzopoulos *et al.*, 1991). Although such models are straightforward to implement in real-time, the absence of internal properties make them unsuitable for creating realistic biomechanical deformations. Yet such models were popular in computer graphics, but the simplicity of these models had to be offset by the skill of the animation specialist to properly control the deformation. Such an approach was used nonetheless in the field of medical simulation (Cover *et al.*, 1993). The authors developed a laparoscopic simulation of endoscopic gall-bladder surgery (cholecystectomy) combining free-form deformations with active surfaces to represent the outcome of the surgical manipulation. Further improvements to the free-form deformation algorithm were proposed (Hirota *et al.*, 1999) to allow better volume preservation characteristics and more intuitive methods of mesh manipulation (Hu *et al.*, 2001).

Spring-mass models

In spring-mass models, deformable objects are approximated by a collection of point masses connected by weightless springs, which are usually damped to control vibration. Real-time computation and topological changes are easily achieved for relatively large models, although subjective testing and an initial processing step is usually required to optimize the choice of parameters (Radetzky *et al.*, 2000) (Hoppe et al., 1993). The main drawbacks of mass-spring systems are related to the topological design and the validity of the deformations. Since springs are used to constrain the distance between two vertices, the number of springs per vertex conditions the global behavior of the system. If the system is under-constrained, several rest positions are possible and the system can fall into unwanted local minima. If the system is over-constrained, the range of deformation is restricted. Also, mass-spring models are not based upon continuum mechanics and cannot model deformations accurately beyond a certain point. For small deformations, a spring model behaves similarly to a linear elastic finite element

model as verified by Keeve *et al.* (Keeve, et al., 1999), but the two methods cannot otherwise be easily compared.

Recently, optimization strategies relying on hardware solutions, in particular recent generations of GPUs, have been used to improve the performance and the quality of simulations based on spring-mass systems. In the context of medical simulation, we can cite the work of (Mosegaard *et al.*, 2005) and (Sørensen *et al.*, 2006). The authors have used an approach based on a spring-mass system to design a simulation of a cardiac procedure, suitable for pre-operative planning. By relying on the impressive computational capabilities of recent GPUs, a 30-fold decrease in solution times was possible, allowing the interactive simulation of models of up to 100,000 nodes. The authors also suggested that optimal spring stiffness parameters can be determined by using an evolutionary algorithm in which the fitness of the mesh was assessed by evaluating the extension of the springs compared with a reference model during the period required for the mesh to return to equilibrium. A static finite element model was chosen as the reference which, although slower, always returned to its undisturbed state. Mass-spring systems, on the other hand, reacted considerably faster but did not reach absolute equilibrium (Mosegaard, 2004).

Yet, the success of this approach certainly comes more from the simplicity of the model than its computation efficiency. We have shown in (Cotin *et al.*, 1999) that the computational cost of an explicit FEM method (using pre-computed tensors) is nearly identical to the spring-mass model, without any of the drawbacks listed above. Despite these drawbacks, numerous commercial systems and research prototypes have been developed using these models.

Voxel-based approaches

Besides reducing objects to their surface geometry, or creating complex FEM meshes, another approach consists in building simple three-dimensional meshes (e.g. composed of cubic elements) over which given properties can be defined. Avila and Sobierajski (Avila *et al.*, 1996) created a voxel-based model which used image intensities (and intensity gradients) to propagate stiffness factors across the mesh. Forces could then be generated to allow the haptic exploration of visualization abstractions such as iso-surfaces. In the “chain mail” algorithm proposed by Gibson *et al.* (Gibson *et al.*, 1998) the deformation information is propagated efficiently across the volume of the object in two-stages. The model relies on a regular grid of voxels, which are linked to their 6 nearest neighbors. When one node of the structure is pulled or pushed, neighboring links absorb the movement by taking up slack in the structure. If a link between two nodes is stretched or compressed to its limit, displacements are transferred to neighboring links. In this way, small displacements of a selected point in a relatively slack system result in only local deformations of the system, while displacements in a system that is already stretched or compressed to its limit causes the whole system to move. Much like the links in a chain, neighbors only respond to a given nodes movement if the constraints on distances between nodes are violated. Tissue elasticity is modeled in a second process that adjusts the distance between local neighboring volume elements to minimize a local energy constraint. During interactions such as tissue manipulation, only the first part of the deformation algorithm takes place, elastic behavior is only enabled once interactions are reduced or stopped. The authors used this model to develop a prototype of simulator for knee arthroscopic surgery.

Hybrid approaches

Several researchers have proposed methods for real-time simulation of deformable structures that do not clearly fall under a specific category. This corresponds to methods that rely on heuristics that are well adapted to a specific problem but are difficult to generalize to other problems. Some other approaches

are really hybrid in the sense that they combine different methods to achieve a particular result. A rather *ad hoc* technique for computing isochoric deformations was proposed in (Balaniuk *et al.*, 2003) and (Costa *et al.*, 2000) with the "Long" and "Radial" Elements, in which artificially constructed long or radial elements are coupled with Pascal's principle to reduce mesh complexity whilst guaranteeing volume conservation. Neither method requires any numerical scheme of integration or pre-computed condensation step and both methods have been successfully tested with haptic simulations of less than 1,000 elements. However, a drawback of this approach is that certain shapes are very difficult to represent with such elements, thus limiting its applicability, particularly in the medical domain.

An example of hybrid approach is used in the KISMET laparoscopic system (Kühnapfel *et al.*, 1995) (Kühnapfel *et al.*, 2000). The system uses mass-spring models for cutting operations and condensed static FEM where accuracy is needed. The nature of the interaction between the two models was however not detailed. This work was very similar to an approach we proposed in (Cotin *et al.*, 2000a) with a hybrid model composed of a pre-computed linear FEM (as introduced above) and a tensor-mass model. The mesh of the deformable structure is decomposed into different regions, some where the pre-computed linear FEM model applies, others where the tensor-mass model applies (see Figure 20).

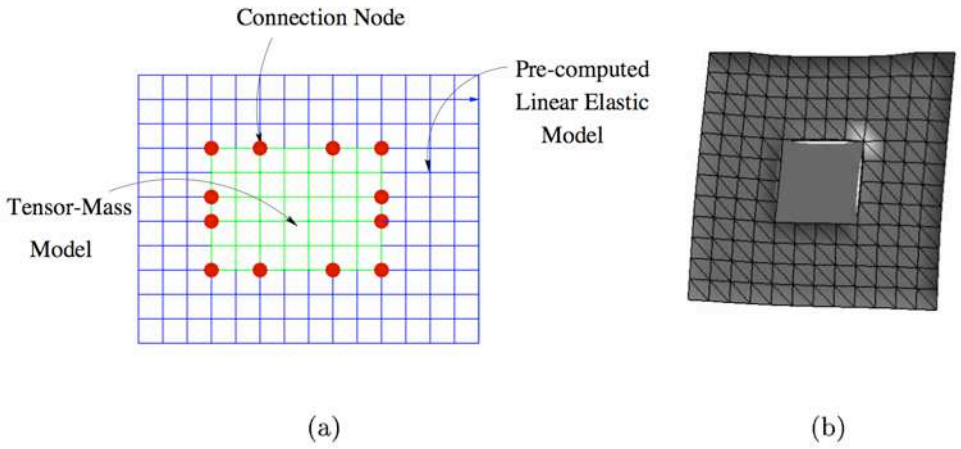


***Figure 20**: Definition of the connection nodes in a hybrid elastic model; (b) Hybrid elastic model deforming under the action of a gravity force.*

The different regions are connected to each other using a set of connection nodes that are described as boundary conditions (Figure 20). Since both models are based on linear elasticity theory, the computed internal forces are identical on either model under the same conditions (Figure 21). This ensures that the global behavior of the deformable structure remains the same regardless of its partitioning.

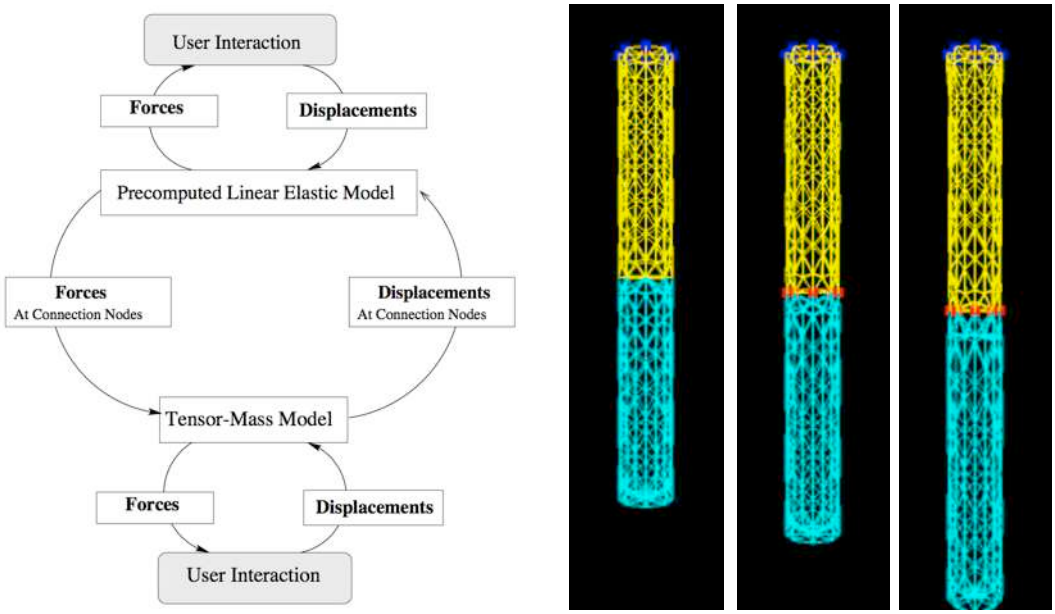


***Figure 21**: (Left): Simulation loop for a hybrid model composed of Pre-computed Linear Elastic model and a tensor mass model. (Right): Example of a deformation using this hybrid representation. Once can see that the deformations are identical in both regions of the mesh if they are of the same size and submitted to the same constraints.*

Since the main benefit of the tensor mass model over the pre-computed FEM model is its ability to perform topological changes, it is possible to create a mesh where a tensor mass model applies only in regions where a cutting operation can occur. This approach was used in a simulation of hepatectomy (Figure 22).

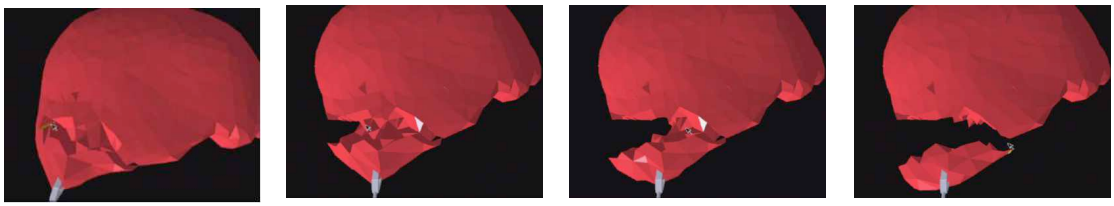


***Figure 22**:* Sequence of images issued from a real-time simulation. The liver model contains about 1,000 nodes. The use of the tensor-mass model allows for simple topological changes (in this case the progressive removal of tetrahedral elements).

Topological changes

Several methods have been proposed to allow topological changes to finite element meshes in real-time. Bielser *et al.* (Bielser *et al.*, 1999) used pre-defined splitting paths to create smooth edges. This method generates a rather larger number of extra tetrahedra for each split tetrahedron and led to further improvements (Bielser *et al.*, 2000). To avoid this issue, Nienhuys *et al.* (Nienhuys *et al.*, 2001) performed cuts along faces of the mesh using a heuristic for determining which faces to cut. Nodes within the mesh were relocated to align the mesh with a virtual scalpel and prevent a jagged surface appearance. Using an optimized conjugate gradient solution, the authors obtained interactive frame rates for a mesh of around 1,500 nodes.

A possible answer for handling topological changes while remaining compatible with continuum mechanics and elasticity theory may lie in the realm of so-called meshless methods. One of the first application of such methods in the context of medical simulation was suggested in (De at al., 2001). Another approach was proposed in (Vigneron *et al.*, 2004) through the extended FEM technique (or XFEM). By incorporating discontinuous functions into the standard FEM, the XFEM can be used to model incisions, without the need for re-meshing. The choice of these functions was explored by the authors and a successful simulation was created using simple two-dimensional objects.

Time integration

The computational efficiency of the various algorithms presented above relies on several aspects. Some methods take advantage of a specific formulation of the constitutive model of the deformable structure, others focus on the optimization of the solver necessary to solve the linear system of equations resulting from the FEM analysis, and finally a few approaches rely on specific choices of the time integration scheme.

In order to simulate dynamic deformable solids, we need to know the time dependent world coordinates $\mathbf{x}(\mathbf{m},t)$ of all points in M. Given $\mathbf{x}(\mathbf{m},t)$ we can subsequently display the configurations $\mathbf{x}(0)$, $\mathbf{x}(\Delta t)$, .. describing the motion and deformation of the object. Here Δt is a fixed time step of the simulation and $\mathbf{x}(t)$ represents the entire vector field at time t. The unknown vector fields $\mathbf{x}(t)$ are not given directly but implicitly as the solution of a differential equation, namely Newton's second law of motion of the form

$$\ddot{\mathbf{x}} = F(\dot{\mathbf{x}},\mathbf{x},t) \quad (29)$$

where $\ddot{\mathbf{x}}$ and $\dot{\mathbf{x}}$ are the second and first time derivatives of $\mathbf{x}$ (i.e. acceleration and velocity) and $F()$ is a general function given by the physical model of the deformable object. In order to find the solution $\mathbf{x}(t)$, this second order differential equation is often rewritten as a coupled set of two first order equations

$$\begin{cases} \mathbf{v} = \dot{\mathbf{x}} \\ \dot{\mathbf{v}} = F(\mathbf{v},\mathbf{x},t) \end{cases} \quad (30)$$

A discrete set of values $\mathbf{x}(0)$, $\mathbf{x}(\Delta t)$, .. of the unknown vector field $\mathbf{x}$ can then be obtained by numerically solving (i.e. integrating) this system of equations. Numerical integration of ordinary differential equations is a problem that has been extensively addressed in various scientific disciplines (see (Hauth *et al.*, 2003) for an overview in the context of deformable modeling in computer graphics). One of the simplest integration schemes is the explicit (or forward) Euler integration, where the time derivatives are replaced by finite differences $\dot{\mathbf{v}}(t)=[\mathbf{v}(t+\Delta t)-\mathbf{v}(t)]/\Delta t$ and $\dot{\mathbf{x}}(t)=[\mathbf{x}(t+\Delta t)-\mathbf{x}(t)]/\Delta t$. Substituting these into the above equations and solving for the quantities at the next time step $t+\Delta t$ yields

$$\begin{cases} \mathbf{x}(t+\Delta t) = \mathbf{x}(t) + \Delta t\ \mathbf{v}(t) \\ \mathbf{v}(t+\Delta t) = \mathbf{v}(t) + \Delta t\ F(\mathbf{v}(t),\mathbf{x}(t),t) \end{cases} \quad (31)$$

Time integration schemes are evaluated by two main criteria, their stability and their accuracy. Their accuracy is measured by their convergence with respect to the time step size Δt , i.e. first order $O(\Delta t)$, second order $O(\Delta t^2)$, etc. In the field of real-time medical simulation, stability is often much more

important than accuracy, since interactions from the user are non predictable and are non smooth. The integration scheme presented above is called explicit because it provides explicit formulas for the quantities at the next time step. Explicit methods are easy to implement but they are only conditionally stable, i.e. stable only if Δt is smaller than a stability threshold. This threshold depends on the stiffness of the object and size of the elements in a FEM mesh. At the exception of very soft objects, this threshold can be quite small. If because of real-time constraints, the actual time step is greater than this threshold, instabilities can appear. These instabilities are due to the fact that explicit methods extrapolate a constant right hand side blindly into the future as the above equations show. For a simple spring and a large Δt, the scheme can overshoot the equilibrium position arbitrarily. At the next time step the restoring forces get even larger resulting in an exponential gain of energy and finally an "explosion". This problem can be solved by using an implicit scheme that uses quantities at the next time step $t+\Delta t$ on both sides of the equation

$$\begin{cases} \mathbf{x}(t+\Delta t) = \mathbf{x}(t) + \Delta t \ \ \mathbf{v}(t+\Delta t) \\ \mathbf{v}(t+\Delta t) = \mathbf{v}(t) + \Delta t \ \ F(\mathbf{v}(t+\Delta t), \mathbf{x}(t+\Delta t), t) \end{cases} \quad (32)$$

The scheme is now called implicit because the unknown quantities are implicitly given as the solution of a system of equations. Now, instead of extrapolating a constant right hand side blindly into the future, the right hand side is part of the solution process. Remarkably, the implicit (or backward) Euler scheme is stable for arbitrarily large time steps Δt. This gain comes with the price of having to solve an algebraic system of equations at each time step (linear if $F()$ is linear, non-linear otherwise). A simple improvement to the forward Euler scheme is to swap the order of the equations and use a forward-backward scheme

$$\begin{cases} \mathbf{v}(t+\Delta t) = \mathbf{v}(t) + \Delta t \ \ F(\mathbf{v}(t), \mathbf{x}(t), t) \\ \mathbf{x}(t+\Delta t) = \mathbf{x}(t) + \Delta t \ \ \mathbf{v}(t+\Delta t) \end{cases} \quad (33)$$

The update of $\mathbf{v}$ uses forward Euler, while the update of $\mathbf{x}$ uses backward Euler. The method is still explicit, $\mathbf{v}(t+\Delta t)$ is simply evaluated first. The forward-backward Euler scheme is more stable than standard forward Euler integration, without any additional computational overhead. There exist a number of other integration schemes, both implicit and explicit (see (Hauth *et al.*, 2003) for instance).

Solvers

Once a time integration scheme has been chosen, a system of equations can then be defined which solution gives the configuration of the deformable body at the next time step. In general, the dimension of the system of equations is directly proportional to the number of degrees of freedom in the system. For instance, a three-dimensional finite element mesh containing N nodes will lead to a system of equations with $3N$ unknowns. The number of nodes required to discretize the shape of a deformable anatomical structure is usually rather large (several thousands to tens of thousands nodes) due to the geometric complexity of most organs. In addition, when using the finite element technique, there is a direct relationship between the size of the elements used in the mesh and the accuracy of the solution (Bathe, 1996). For a problem without singularities, and assuming linear elements are used, the asymptotic error for a mesh with largest element size h is $\delta_h = O(h)$. This essentially means that a large number of elements (and nodes) is often required to discretize anatomical structures and compute their deformable

behavior. This leads to very large systems of equations, and solving such systems in real-time is a very challenging task.

Several approaches have proposed specific time integration strategies that are computationally efficient yet stable. For instance, implicit integration schemes, although more computationally expensive, allow the use of larger time steps (Baraff, *et al.*, 1998). On the other hand, when interacting with soft objects (as it is often the case in medical procedures), explicit integration schemes can be used (Delingette *et al.*, 1999), (Zhuang *et al.*, 2000) (Irving *et al.*, 2004) as long as the time step is "sufficiently small".

Future directions

Modeling soft-tissue deformations remains one of the key challenges in medical simulation. The past ten years have seen important developments in this area, not only in terms of computational performance but also in terms of modeling accuracy. Yet, with the recent developments in the area of patient-specific simulation and planning, more accurate biomechanical models need to be developed. Unfortunately, as more accurate models will be developed, they will generally become more specific to a particular type of tissue or organ. Developing more accurate soft tissue models will also require a better understanding and characterization of the real world. This characterization, as we have seen in this section, involves three major steps: the choice of a constitutive model that characterizes a particular behavior, assumptions about the model or its applicative context to simplify the equations, and the identification of the model parameters based on experimental or patient-specific measurements. Of course, once a model has been defined, numerical techniques for solving the equations associated with the constitutive model need to be developed. The choice of an implementation and even the hardware on which this implementation will run, also play a role in the overall results. Finally, the resulting model has to be validated against new experimental data to measure its accuracy and predictive capability.

There are, however, several limitations to the development of new biomechanical models of soft-tissues. First, acquiring experimental data, in *in vivo* or near *in vivo* conditions is challenging and time consuming, as we have seen in this section. Also, current experimental modalities rely essentially on one-dimensional data (stress-strain curves) while the phenomena we are modeling are principally three-dimensional. The addition of imaging techniques to help characterize and validate models seems a direction of interest. Imaging techniques could also play a key role in identifying patient-specific parameters for a given biomechanical model. Enabling patient-specific simulation will only be possible by providing geometrical, biomechanical, and potentially physiological information about the patient. Imaging techniques such as elastography could be a direction where to start investigating.

As we start increasing the complexity of biomechanical models, their computational requirements will dramatically increase. In particular, a lot of work remains to be done for solving large non-linear models in real-time. Similarly, very few works have addressed the problem of topological changes, which remains a key aspect of a large number of medical procedures. Yet, it is obvious that we will soon reach a point where solutions only based on intelligent algorithmic will not be sufficient for reaching real-time computation. The solution is then to rely more on the hardware, as shown in their pioneering work by Szekely *et al.* (Szekely *et al.*, 2000) who employed massively parallel hardware to produce a fully dynamic, non-linear simulation of endoscopic procedures. Current generations of GPUs could certainly offer some interesting, and low cost, alternatives to traditional clusters or parallel machines. Other directions we plan to explore in our research include meshless techniques and multi-scale approaches (through hierarchical meshes and hierarchical basis functions).

Summary of contributions

- **Research articles**
  - S. Cotin, H. Delingette, N. Ayache. “Real-time elastic deformations of soft tissue for surgery simulation”. IEEE Transactions on Visualization and Computer Graphics 1999 ; 5(1) : 62-73.
  - J. Marescaux, J.-M. Clement, V. Tassetti, C. Koehl, S. Cotin, Y. Russier, D. Mutter, H. Delingette, N. Ayache. “Virtual reality applied to hepatic surgery simulation: The next revolution”. Annals of Surgery 1998, 228(5) :627-634.
  - N. Ayache, S. Cotin, H. Delingette, J.-M. Clement, J. Marescaux, M. Nord. “Simulation of Endoscopic Surgery”. Journal of Minimally Invasive Therapy and Allied Technologies 1998; 7(2): 71-77.
  - H. Delingette, S. Cotin and N. Ayache. “A Hybrid Elastic Model Allowing Real-Time Cutting, Deformations and Force-Feedback for Surgery Training and Simulation”. Computer Animation 1999, May 1999, Geneva, Switzerland.
  - N. Ayache, S. Cotin, and H. Delingette. “Surgery Simulation with visual and haptic feedback”. In Y. Shirai and S. Hirose, editors, Robotics Research, the Eighth International Symposium, pages 311-316. Springer, 1998.
  - S. Cotin, H. Delingette, J.-M. Clément, M. Bro-Nielsen, N. Ayache, and J. Marescaux. “Geometrical and Physical Representations for a Simulator of Hepatic Surgery”. Proc. Medicine Meets Virtual Reality IV (MMVR' 96), 1996, pp. 139-150.
  - S. Cotin, H. Delingette, and N. Ayache. “Real-time volumetric deformable models for surgery simulation”. In Visualization in Biomedical Computing, Proceedings, volume 1131, Lecture Notes in Computer Science. Springer Verlag, New York, 1996.
  - M. Bro-Nielsen and S. Cotin. “Real-time volumetric deformable models for surgery simulation using finite elements and condensation”. Eurographics'96, Computer Graphics Forum 1996; 15(3) :57-66.
- **Patents**
  - Cotin S, Delingette H, Ayache N; "Electronic Device for processing image-data for simulating the behavior of a deformable object", International patent WO9926119A1, 1998.
- **Software**
  - **Real-time FEM code**: code developed during my PhD thesis; development time: 30 months; more than 20,000 lines of code; used in the development of a prototype of a training system for laparoscopic surgery with force feedback.
  - **Training system for laparoscopic surgery**: a prototype of a simulation system, based on my work on real-time deformable models, was developed. It served for many demonstrations, as a basis for discussions with physicians, and was featured on many occasions in scientific popularizing programs.
- **Miscellaneous**
  - The work on laparoscopic surgery simulation was featured at several occasions on national TV programs, including national news (France 3 in 1996), a national TV Health Program (Santé à la Une in 1997). It was also featured on a national radio program (France Info, 1997). I was also featured in VSD Magazine in a 3 page report. The article was entitled "Opérer Comme Dans Un Jeu Vidéo" (VSD, November 1996, pp. 26-28).

4. Physiological Modeling

Physiology has traditionally been concerned with the integrative function of cells, organs and whole organisms. Today, biologists are essentially focusing on the understanding of molecular and cellular mechanisms, and it becomes increasingly difficult for physiologists to relate whole organ function to underlying cellular or molecular mechanisms. Organ and whole organism behaviors need to be understood at both a systems level and in terms of cellular functions and tissue properties. While mathematical and computational modeling have the potential to cope with complex engineering problems, biological systems, however, are vastly more complex to model. We have seen a similar situation through the section on Biomechanical Modeling. Understanding and modeling physiology as a whole is even more complex, as biomechanical modeling of organs can be seen as only a subset of physiological modeling. At the exception of the recently started Physiome Project[3] there has never been a proposal for developing multi-scale models unifying the various aspects of physiological modeling. For all these reasons, physiology remains an aspect of computer-based medical simulation that has very rarely been tackled.

Mannequin-based training systems are the only ones to rely heavily on physiological models, although in this case the models are very simple. Usually the underlying representation is a lumped-parameter approximation of cardiovascular hemodynamics. These systems, since they often aim at anesthesiology training, also include simplified pharmaceutical models that control the influence of a series of drugs on the hemodynamic system (Eason *et al.*, 2005). Regarding computer-based simulators, very few of the existing systems incorporate physiological models, and those are essentially simplified hemodynamic models (Masuzawa *et al.*, 1992) (Dawson *et al.*, 2000). Additionally, some work on tumor growth and angiogenesis was proposed recently. The algorithms do not necessarily allow real-time computation but are less computationally expensive than traditional models. For instance, Clatz *et al.* (Clatz *et al.*, 2004) proposed a model for simulating the growth of glioblastoma multiforma, a glial tumor. They used the finite element method to simulate both the invasion of tumors in the brain parenchyma and its interaction with the invaded structures. The former effect is modeled with a reaction-diffusion while the latter is based on a linear elastic brain constitutive equation. The mechanical influence of the tumor cells on the invaded tissues was also taken into account. Lloyd *et al.* (Lloyd *et al.*, 2007) present a model of solid tumor growth which can account for several stages of tumorigenesis, from the early avascular phase to the angiogenesis driven proliferation. The model combines several previously identified components in a consistent framework, including neoplastic tissue growth, blood and oxygen transport, and angiogenic sprouting. As a final example, the work of Tuchschmid *et al.* (Tuchschmid *et al.*, 2007) concerns the simulation of intravasation of liquid distention media into the systemic circulation as it occurs during hysteroscopy. A linear network flow model is extended with a correction for non-newtonian blood behavior in small vessels and an appropriate handling of vessel compliance. Cutting of tissue is accounted for by adjusting pressure boundary conditions for all cut vessels while real-time simulation is possible by using a fast lookup scheme for the linear model. There is certainly a larger body of work that could be relevant to interactive medical simulation than the one highlighted here. Nonetheless, it is clear that the problems of hemodynamics, arterial or venous flow, bleeding, altered organ functions, etc. have been less studied in the context of interactive simulation than soft tissue deformation.

Of equal importance than the role of physiological models is their integration within the simulation system. For instance, several simulations of laparoscopic surgery have proposed models for describing

[3] The "IUPS Physiome Project" is an international collaborative effort to build model databases and computational tools in order to facilitate the understanding of physiological functions in healthy and diseased mammalian tissues by developing a multi-scale modeling framework that can link biological structure and function across all spatial scales.

bleeding, when an artery is cut for instance, but the bleeding has no impact on the other elements of the simulation. It remains essentially a visual effect. To be realistic, from a physiological standpoint, we need to work toward “closed-loop physiology”, i.e. physiological models that, on one level describe a specific behavior, and at a higher level describe global, systemic physiological responses. If we take our previous example, when cutting an artery during surgery, the simulation should compute the corresponding change in blood pressure, the corresponding increase in heart rate (necessary to maintain blood pressure), and the impact this increased blood pressure will have on other simulated functions. This forms a closed system which can be broken at some point if the bleeding is not stopped, leading to major changes in patient or organ behavior, and potentially death.

In an attempt to bring more integrated physiological models into real-time simulations, we have proposed a vascular flow model in the context of interventional neuroradiology. This model describes arterial and venous blood flow, its reaction to changes in heart rate as well as local changes in the vascular tree due to a simulated therapeutic procedure. The model also interacts with contrast agent being injected in the blood stream. This work was published in 2007 in the proceedings of the MICCAI conference (Wu *et al.*, 2007) and is part of a larger project, started in 2005 at CIMIT, which aims at producing a high-fidelity training and planning system for interventional radiology (see Chapter V).

Real-time Angiography Simulation

Interventional neuroradiology is a growing field of minimally invasive therapies that includes embolization of aneurysms and arterio-venous malformations, carotid angioplasty and carotid stenting, and acute stroke therapy. Treatment is performed using image-guided instrument navigation through the patient's vasculature and requires intricate combination of visual and tactile coordination. A first but important step in the training curriculum consists in performing diagnostic angiography. An angiography is used to locate narrowing, occlusions, and other vascular abnormalities. By visualizing and measuring flow distributions in the vicinity of a lesion, angiographic studies play a vital role in the assessment of the pre- and post-operative physiological states of the patient. To simulate angiographic studies with a high degree of fidelity, we proposed in (Wu *et al.*, 2007) a series of techniques for computing, in real-time, blood flow and blood pressure distribution patterns, as well as the mixing of blood and contrast agent.

Real-time flow computation in large vascular networks

The first step toward real-time physiologic representations of arterial, parenchymal and venous phases of thoracic, cervical and intracranial vasculature is the choice of a flow model. Although turbulent flow patterns are visible in ventriculograms, in some aortic angiograms or near aneurysms, diagnostic angiography rarely depends on such data. Instead, flow distribution in the network is more relevant when identifying and quantifying vessel pathology than local fluid dynamics pattern. Hence, a one-dimensional laminar flow model is adequate for a majority of applications, and particularly when considering the cerebrovascular system, where blood velocity is limited and vessels are very small. Blood flow in each vessel is modeled as an incompressible viscous fluid flowing through a cylindrical pipe, and can be calculated from a simplified Navier-Stokes equation called Poiseuille’s Law (Eq. 34). This equation relates the vessel flow Q to the pressure gradient ΔP, blood viscosity η, vessel radius r, and vessel length L

$$Q = \frac{\pi r^4 \Delta P}{8 \eta L} \quad (34)$$

To compute such vascular flow, a finite element model was developed based on the observation that the arterial (or venous) system can be represented as a directed graph, with M edges and N nodes. If $M \neq N$ an augmented square matrix $\mathbf{K}$ is formed by adding trivial equations, i.e. $P_s = P_s$ or $Q_s = Q_s$ to the set of

Poiseuille equations. Assuming (for sake of simplification) that $M<N$ we can write the following system of equations in matrix form

$$\mathbf{Q} = \mathbf{KP} \quad \text{or} \quad \begin{bmatrix} \vdots \\ Q_i \\ \vdots \\ \hline P_s \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \Omega_i & 0 & -\Omega_i & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \hline 0 & \cdots & \cdots & 0 & I \end{bmatrix} \begin{bmatrix} \vdots \\ P_j \\ \vdots \\ P_k \\ \vdots \end{bmatrix} \quad (35)$$

where $\Omega_i=1/R_i$ is the vessel i flow resistance.

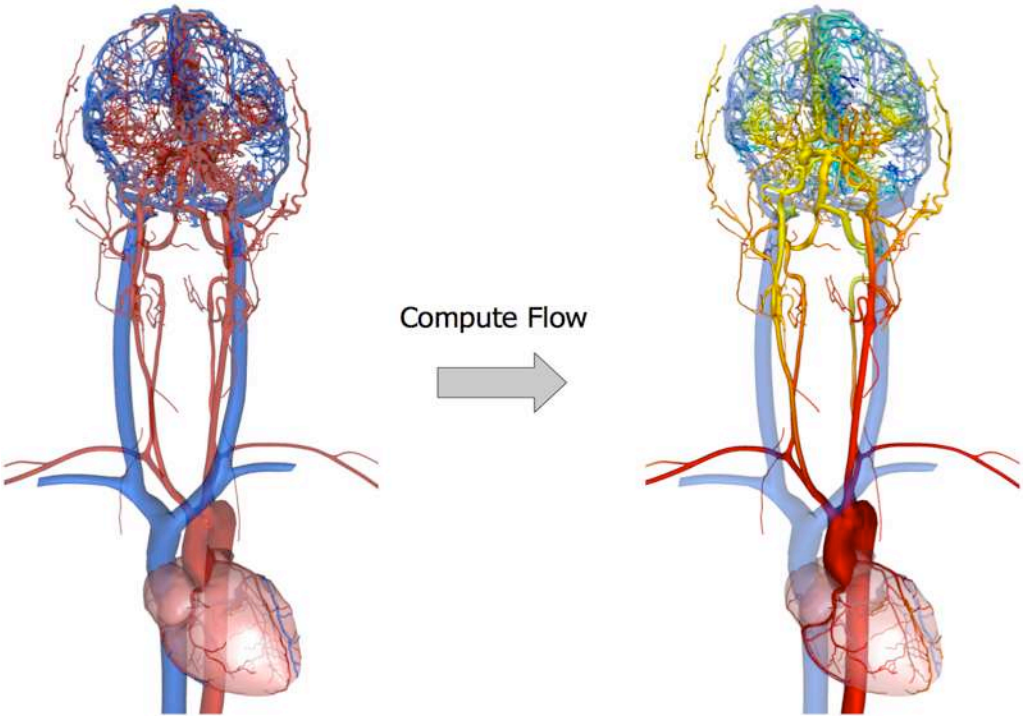


***Figure 23**: Blood flow and pressure distribution in the cerebrovascular system. The arterial vascular network is composed of more than 3,000 vessels, yet the computation is performed in real-time.*

Boundary conditions, corresponding to 2 types of constraints, are then added to the system as Lagrange Multipliers. The first set of constraints corresponds to the prescribed pressure values at the beginning and end nodes of the directed graph. These pressure values are defined as a function of time, depth of the node in the tree graph, and ventricular pressure. Time-varying (and cyclic) cardiac pressure can then be set at the aortic root, while nodes that are deep into the graph are assigned a smaller pressure, which is also less variant in time. The second set of boundary conditions relates to the conservation of flow, similar to Kirchhoff's circuit laws in electric circuits, such that for any internal node, the total flow flowing toward this node is equal to the total flow flowing away from this node. This is described by

$$\mathbf{0} \equiv \sum \mathbf{Q}_{in} + \sum \mathbf{Q}_{out} = \mathbf{\Psi}^T \mathbf{P} \quad (36)$$

where $\mathbf{\Psi}^T$ is a matrix in which each row is a summation of multiple rows of $\mathbf{K}$. The resulting system matrix of the one-dimensional flow FEM model is

$$\bar{\mathbf{Q}} \equiv \begin{bmatrix} \mathbf{Q} \\ \mathbf{P}_e \\ \mathbf{0} \end{bmatrix} = \bar{\mathbf{K}}\bar{\mathbf{P}} = \begin{bmatrix} \mathbf{K} & \Gamma & \Psi \\ \Gamma^T & 0 & 0 \\ \Psi^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P} \\ \lambda_e \\ \lambda_f \end{bmatrix} \quad (37)$$

where λ_e and λ_f are vectors of Lagrange multipliers, and $\mathbf{\Gamma}$ is a permutation matrix. $\mathbf{P_e}$ is a vector containing preset pressures. Since the pressure values P_e vary in time, the vector $\bar{\mathbf{Q}} = \begin{bmatrix} \cdots 0 & \cdots & \mathbf{P}_e^T 0 \end{bmatrix}^T$ containing both initial conditions $\mathbf{Q}=\mathbf{0}$ and boundary conditions, needs to be updated at every time step. Given $\bar{\mathbf{Q}}$ we can then compute $\bar{\mathbf{P}} = \bar{\mathbf{K}}^{-1}\bar{\mathbf{Q}}$ and then $\mathbf{Q} = \begin{bmatrix} \mathbf{K} & \mathbf{G} & \Psi \end{bmatrix}\bar{\mathbf{P}}$. Assuming the vascular resistance is invariant in time, $\bar{\mathbf{K}}^{-1}$ can be pre-computed.

This permits real-time computation rates, even on very large vascular models, comprising several thousands arteries and veins. For instance, on a vascular data set containing 2,337 arteries, the flow distribution was computed in about 5 milliseconds, on a Pentium Dual Core 1.6 GHz (Figure 23). It is important to remember that this simulation also accounts for time-varying blood pressure at the aortic root, thus providing a global and realistic flow pattern across the arterial cerebrovascular system.

Modeling local blood flow alterations

For the purpose of training, it is important to have access to a variety of scenarios. Each scenario does not necessarily need to be based on an existing patient data set, but could be derived from a generic data set that is altered to create a specific pathology. Additionally, during stenting or angioplasty simulation, a stent or balloon is deployed to expand a narrowed section of a vessel, called a stenosis (see Figure 24). As the vessel's radius changes during a procedure, its resistance also changes according to Poiseuille's law. This change in resistance results in a change in blood flow, not only through the vessel but throughout the vascular system. This change in flow needs to be computed in real-time.

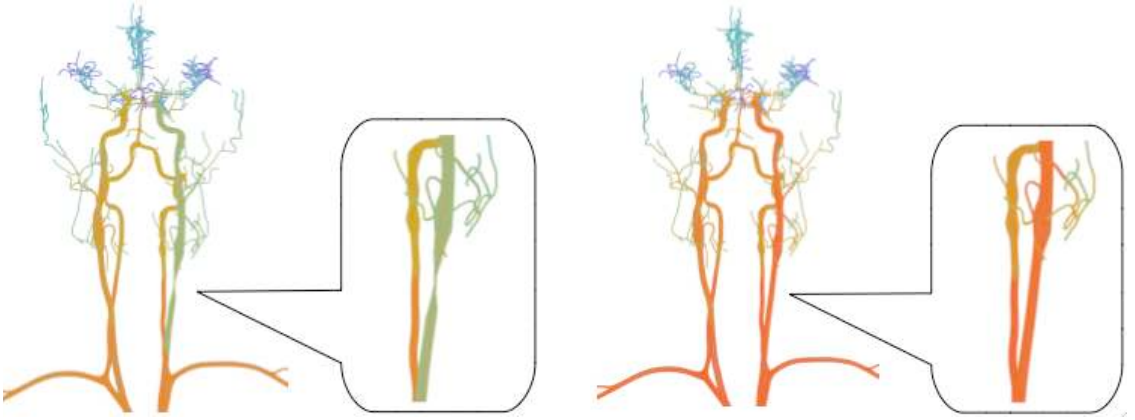


***Figure 24**: Pressure distribution in the arterial cerebrovascular system for both a normal patient and a patient with a simulated stenosis in the common carotid artery.*

Assuming vessel i contains a stenosis, we can decompose its shape in three sections, one corresponding to the narrowed part of the vessel, with resistance R_S, one before the stenosis with resistance R_1, and one after the stenosis with resistance R_2. Thus the initial resistance of the vessel is $R_i = R_1 + R_2 + R_S$ and its resistance after the vessel's expansion becomes $R'_i = R_i + (R'_s + R_S)$ where R'_s is the new radius of

the previously narrowed section. This change in resistance requires to update $\overline{K}$. Let $\overline{K'}$ be the updated $\overline{K}$. The inverse $\overline{K}^{'-1}$ of $\overline{K'}$ needs to be computed efficiently to maintain real-time performances. Since the update of $\overline{K}$ is only local, namely row i in $\overline{K}$ and two rows in Ψ^T, $\overline{K}$ can be rewritten as

$$\bar{\mathbf{K}}' = \bar{\mathbf{K}} + \mathbf{U}\mathbf{V}^T \text{ with } \begin{cases} \mathbf{U} = \begin{bmatrix} \cdots & 1 & \cdots & 1 & \cdots & -1 & \cdots \end{bmatrix}^T \\ \mathbf{V}^T = \begin{bmatrix} \cdots & \Delta\Omega_i & \cdots & -\Delta\Omega_i & \cdots \end{bmatrix}^T \end{cases} \quad (38)$$

where $\mathbf{U}$ and $\mathbf{V}$ are dimensionally compatible matrices, and $\Delta\Omega_i = R_s' - R_s$ represents the weight change for each of the end nodes of vessel i. From this, the new inverse $\overline{K}'^{-1}$ can be efficiently computed using Woodbury's formula (Golub, 1996).

Similarly, when a contrast agent is injected within the vascular network to enhance the contrast of blood vessels in fluoroscopic images, the added fluid changes the flow rate near the point of injection (the tip of the catheter). We model its local influence on the vascular flow by modulating the flow rate in the vessel by the injection rate. This is achieved through an additional boundary condition

$$Q_{inj} = (P_{i,0} - P_{i,1}) / R_i = \begin{bmatrix} \cdots & \Omega_i & \cdots & -\Omega_i & \cdots \end{bmatrix} \overline{P} = U_{inj}{}^T \overline{P} \quad (39)$$

such that

$$\bar{\mathbf{K}}_{inj} = \begin{bmatrix} \bar{\mathbf{K}} & \mathbf{U}_{inj} \\ \mathbf{U}_{inj}^T & 0 \end{bmatrix} \quad (40)$$

To achieve real-time performance, $\bar{\mathbf{K}}_{inj}^{-1}$ must be computed efficiently as well. Given $\bar{\mathbf{K}}^{-1}$ this is achieved by using a block matrix decomposition.

Authors	Flow in CCA	Flow in ICA	Flow in VA
Schoning *et al.*	470±120 ml/min	265±26 ml/min	85±33 ml/min
Weskott *et al.*	417±87 ml/min	N/A	N/A
Seidel *et al.*	N/A	N/A	91±37 ml/min
Dorfler *et al.*	N/A	238±45 ml/min	82±44 ml/min
Yazici *et al.*	418±100 ml/min	231±59 ml/min	85±37 ml/min
Our approach	**433** ml/min	**240** ml/min	**86** ml/min

***Table 1**: comparative table of blood flow measurements or computations reported in the literature. We can see that our real-time model achieves good accuracy in the different sections of the vascular tree we information is available: Common Carotid Artery (CCA), Internal Carotid Artery (ICA) and Vertebral Artery (VA).*

Advection-diffusion model

An angiography is done by taking a continuous series of X-rays while injecting a contrast agent into the vascular structure under examination. The contrast agent, usually an iodine solution, provides the density needed for detailed X-ray study of the blood vessels. Upon injection, the contrast agent is carried by blood cells and circulates through the vascular system (arterial and venous) until it is eliminated in the kidneys and liver. We model the transportation of contrast agent by an advection-diffusion equation describing the distribution of contrast agent concentrations $C(x,r,t)$ as a function of curvilinear coordinates x along the centerline of a vessel, the distance r to the centerline, and time t as

$$\frac{\partial C(x,r,t)}{\partial t} + u(x,r,t)\frac{\partial C(x,r,t)}{\partial x} + \kappa_C \nabla \cdot \nabla C(x,r,t) = I(x,t) \qquad (41)$$

where $I(x,t)$ is the injection rate of contrast agent, κ_c is the contrast agent diffusion factor and $u(x,r,t)$ is the laminar flow velocity along the axial direction of each vessel. This velocity can be modeled as a parabolic profile

$$u(x,r,t) = -\frac{1}{4\eta}\frac{\Delta P}{\Delta x}(R^2 - r^2) = u_{max}(x,t)\left(1 - \frac{r^2}{R^2}\right) \qquad (42)$$

where $u_{max}(x,t)$ is the velocity at the centerline of the vessel, as computed from the vascular flow model. Compared to only modeling concentration at the centerline, as in most previous works, this model provides two important features: the propagation front is not flat, and a fraction of the contrast agent remains for a longer time in vessels due to the low velocity near the vessel wall. To improve computational efficiency, we made the following simplifications:

- Near the center of the vessel, where the velocity is highest, the advection term is stronger than the diffusion, and this term can be neglected. We solve this advection numerically using an unconditionally-stable semi-implicit integration scheme, introducing some diffusion due to numerical dissipation.
- Near the vessel wall, as the velocity drops to zero, only the diffusion term of (Eq. 41) is relevant. This diffusion can be computed numerically by combining the value from the previous timestep with the concentration at the center, using coefficients that can be derived from the original flow diffusion factor.

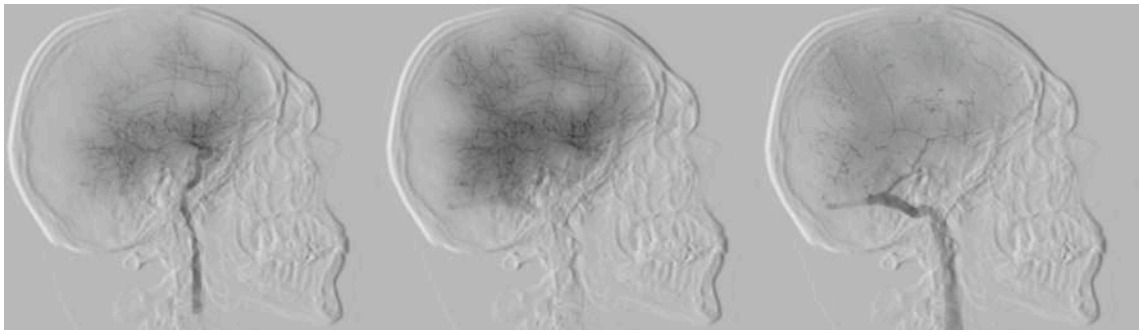


***Figure 25**: High-fidelity real-time simulation of angiography in the brain, featuring contrast injection in arterial flow (Left); blush (Middle); and transition to venous side (Right).*

Simulated angiography

We have applied the models introduced previously to different data sets, and performed both qualitative and quantitative validations. Blood flow and blood pressure distributions were computed on a data set containing about 500 arterial vessels representing the cerebrovascular system, and on a data set containing about 4,000 vessels (arterial and venous) describing the full vascular circulation system with a higher level of detail in the brain. Figure 25 and Figure 26 show two angiograms computed in real-time. They are visually very similar to actual cerebral angiograms, and the kinematic characteristics of contrast agent and blood motion closely match real angiographic data. Various simulations are possible, including selective angiograms and parenchymal blush, and all have been qualitatively validated by the neuro-interventional radiologist working with us on the project.

In addition to the qualitative assessment of the visual quality of the angiograms, a quantitative validation was performed by comparing flow values to results from Yazici *et al.* (Yazici, 2005) as well as other studies referenced in (Yazici, 2005). Results of the quantitative assessment are illustrated in (Table 1) and show a very good adequacy between flow velocities computed with our model and values reported in the literature. Changes in local flow patterns due to the treatment of a stenosis, illustrated in Figure 24, are also realistic enough to provide a high level of fidelity in the training.

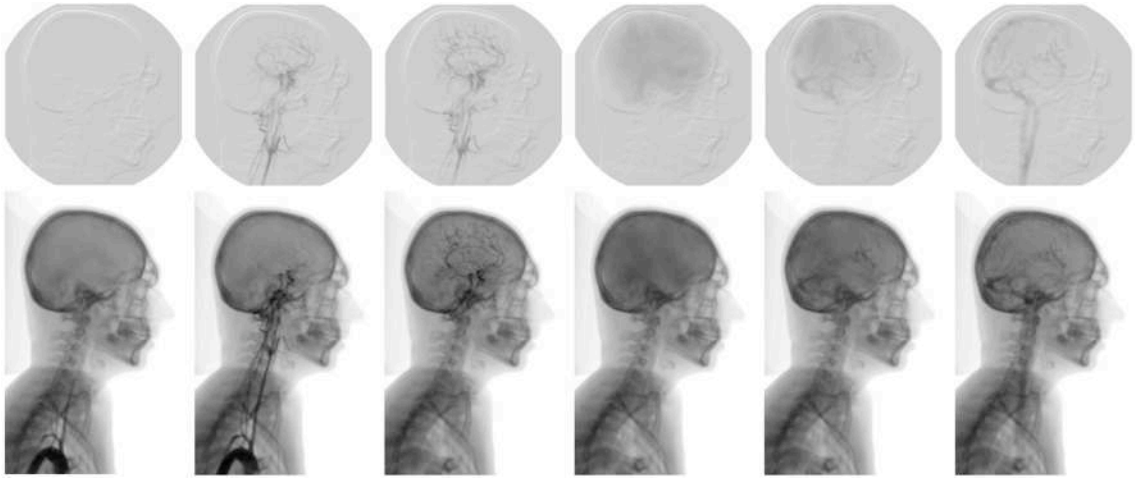


***Figure 26**: Simulated angiographic study showing different steps of the contrast agent propagation. From left to right, the contrast agent, carried by the blood flow, moves through the vascular system, then reaches the brain parenchyma where it diffuses (blush), and then reaches the venous system. The visualization combines volume-rendering techniques of the patient's anatomy and particle-based rendering for the contrast agent.*

Conclusion

The same principles we have previously illustrated apply to physiological modeling, even at the "simple" level we are considering in medical simulation. Physiological modeling should be aimed at understanding and predicting physiological mechanisms, and to be self-consistent and predictive, the proposed models must be built from underlying biophysical principles. In addition, models should be able to link with other models to create more complete, integrated simulations. Our contributions in this area are a step in this direction, but a lot remains to be done to achieve simulations where patient-specific physiological characteristics could be taken into account. This is particularly essential to model the outcome of a procedure, whether it is successful or not, as possibility to err is a key aspect of any simulation.

Summary of contributions

- **Research articles**
  - X. Wu, J. Allard, and S. Cotin. “Real-time Modeling of Vascular Flow for Angiography Simulation”. Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), Volume 4792, pp. 850-857, 2007.
  - J. Rabinov, S. Cotin, J. Allard, J. Dequidt, J. Lenoir, V. Luboz , P. Neumann, X. Wu, and S. Dawson. “EVE: Computer Based Endovascular Training System for Neuroradiolgy”. ASNR 45th Annual Meeting & NER Foundation Symposium, pp. 147-150, 2007.
- **Software**
  - **EVE: interventional radiology training system**: the real-time one-dimensional flow algorithm and the advection-diffusion model have been integrated within a prototype of training system for interventional neuro-radiology.
- **Miscellaneous**
  - Multiple demonstrations of this work were performed during scientific conferences (ATACCC, MMVR, MICCAI) and clinical meetings (ASNR, SIR annual meeting).

5. Medical device modeling

Most publications in the field of medical simulation have addressed issues related to laparoscopic surgery simulation, with a particular focus on the modeling of soft tissue deformation. This was motivated by the need to train surgeons (novices or experienced) to this new technique. At the same time, the technique of laparoscopic surgery itself made it more suitable for simulation. In particular, the manipulation of the tissues, using long instruments transmitting reduced haptic feedback, made it possible to propose plausible simulation techniques. The recent development of new minimally-invasive techniques, such as interventional radiology or natural orifice transluminal endoscopic surgery, have changed the requirements of simulation systems. One of these requirement, which we will address throughout this section, is the modeling of flexible medical devices.

Flexible mechanical devices

Unlike laparoscopic surgery, Natural Orifice Transluminal Endoscopic Surgery (NOTES) avoids the need for abdominal incisions. In NOTES procedures, a flexible endoscope is passed through a natural orifice such as the mouth or rectum, and intra-abdominal procedures are performed through a transvisceral incision. Flexible instruments such as graspers, needles, etc. are then inserted through the lumen of the endoscope into the abdominal cavity. Similarly, interventional radiology requires the manipulation of flexible devices such as catheters and guidewires that are navigated within the vascular network toward a lesion such as a stenosis or an aneurysm. Additional flexible devices, such as coils, can then be deployed.

These various medical devices, although rather different in shape and size, share one characteristic: their behavior can essentially be described by the deformation of a curve corresponding to the centerline of the device. We use the term *wire-like structure* to categorize such devices. The main characteristics of these wire-like structures include geometric non-linearities, high tensile strength and low or controllable resistance to bending. We initially proposed a multi-body dynamics model (Cotin *et al.*, 2000c) where a set of rigid elements are connected using spherical joints (Featherstone, 1983), thus mimicking the basic behavior of such devices. However, such an approach is limited in describing the bending and twisting properties of such devices. Another interesting approach to modeling wire-like structures was introduced in (Lenoir *et al.*, 2002). The proposed model was based on one-dimensional dynamic splines, providing a continuous representation of a deformable currve. Different constraints could be defined to control the model, such as sliding through fixed locations in space. Compatible with real-time computation requirements, this model did not, however, incorporate torsional energy terms. A virtual catheter based on a linear elasticity and beam elements, was introduced by Nowinski *et al.* (Nowinski *et al.*, 2001). The choice of beams for the catheter model is natural since beam equations include cross-sectional area, cross-section moment of inertia, and polar moment of inertia, allowing solid and hollow devices of various cross-sectional geometries and mechanical properties to be modeled. The main limitations were the inability of the model to represent the geometric non-linearities necessary to characterize such flexible devices. Another approach also targeted at virtual catheter or guidewire modeling was proposed by Alderliesten (Alderliesten, 2004). In this model, only bending energies are computed, assuming no elongation and perfect torque control. The model has characteristics similar to a multi-body dynamics model but integrates more complex bending energies, as well as local springs for describing the intrinsic curvature of the catheter. Although based on a more *ad hoc* representation, a good level of accuracy is obtained using this model. The main drawbacks are how collision response is handled during contact with the walls of the vessel, and computation times that are not compatible with real-time requirements.

To improve the accuracy of previously proposed models, and handle geometric non-linearities while maintaining real-time computation, we have developed a new mathematical representation based on three-dimensional beam theory. A first proposition was described in (Cotin *et al.*, 2005) and (Duriez *et al.*, 2006a). The model relies on a static formulation of the deformable behavior. Combined with an incremental approach allowing for geometric non-linearities and a substructure analysis for fast computation times, it allows interactive navigation of virtual devices. A second model was proposed later, using a dynamic formulation and a non-linear approach similar to a co-rotational method (Dequidt *et al.*, 2008). The first model was used to describe catheter and guidewire deformation while the second one was applied to embolization coils.

Incremental formulation

To model the deformation of a catheter, guidewire, or any solid body whose geometry and mechanical characteristics are similar to a wire, rod or beam, we use a representation based on three-dimensional beam theory (Przemieniecki, 1968), where the elementary stiffness matrix $\mathbf{k}_e$ is a 12×12 symmetric matrix that relates angular and spacial positions of each end of a beam element to the forces and torques applied to them:

$$\mathbf{k}_e = \frac{E}{L}\begin{bmatrix}
A & & & & & & & & & & & \\
0 & \frac{12I_z}{L^2(1+\phi_y)} & & & & & & & & & & \\
0 & 0 & \frac{12I_y}{L^2(1+\phi_z)} & & & & & & & & & \\
0 & 0 & 0 & \frac{GJ}{E} & & & & & & & & \\
0 & 0 & \frac{-6I_y}{L(1+\phi_y)} & 0 & \frac{(4+\phi_z)I_y}{1+\phi_z} & & & & & & & \\
0 & \frac{6I_z}{L(1+\phi_y)} & 0 & 0 & 0 & \frac{(4+\phi_y)I_z}{1+\phi_y} & & & & & & \\
-A & 0 & 0 & 0 & 0 & 0 & A & & & & & \\
0 & \frac{-12I_z}{L^2(1+\phi_y)} & 0 & 0 & 0 & \frac{-6I_z}{L(1+\phi_y)} & 0 & \frac{12I_z}{L^2(1+\phi_y)} & & & & \\
0 & 0 & \frac{-12I_y}{L^2(1+\phi_z)} & 0 & \frac{6I_y}{L(1+\phi_y)} & 0 & 0 & 0 & \frac{12I_y}{L^2(1+\phi_z)} & & & \\
0 & 0 & 0 & \frac{-GJ}{E} & 0 & 0 & 0 & 0 & 0 & \frac{GJ}{E} & & \\
0 & 0 & \frac{-6I_y}{L(1+\phi_y)} & 0 & \frac{(2-\phi_z)I_y}{1+\phi_z} & 0 & 0 & 0 & \frac{-6I_y}{L(1+\phi_y)} & 0 & \frac{(4+\phi_z)I_y}{1+\phi_z} & \\
0 & \frac{6I_z}{L(1+\phi_y)} & 0 & 0 & 0 & \frac{(2-\phi_y)I_z}{1+\phi_y} & 0 & \frac{-6I_z}{L(1+\phi_y)} & 0 & 0 & 0 & \frac{(4+\phi_y)I_z}{1+\phi_y}
\end{bmatrix} \quad \text{Symmetric} \tag{43}$$

with $G = E/(2+2v)$ where E is the Young's modulus and v is the Poisson's ratio; A is the cross-sectional area of the beam, and L its length; I_y and I_z are cross-section moments of inertia; Φ_y and Φ_z represent shear deformation parameters and are defined as $\Phi_y = 12EI_z/GA_{sy}L^2$ and $\Phi_z = 12EI_y/GA_{sz}L^2$ with A_{sy} and A_{sz} the shear area in the the y and z directions.

To determine the stiffness of the complete structure, a common reference frame must be established for all unassembled structural elements. Thus the displacements and their corresponding forces will be referred to a common (global) coordinate system. Since the stiffness matrix $\mathbf{k}_e$ is initially calculated in local coordinates, oriented along the frame of first node of the beam (to minimize the computing effort), it is necessary to introduce transformation matrices changing the frame of reference from a local to a global coordinate system. This relationship is expressed by the matrix equation

$$\mathbf{K}_e = \Lambda^T \mathbf{k}_e \Lambda \quad (44)$$

where Λ is a matrix of coefficients obtained from the direction cosines of angles between the local and global coordinate systems.

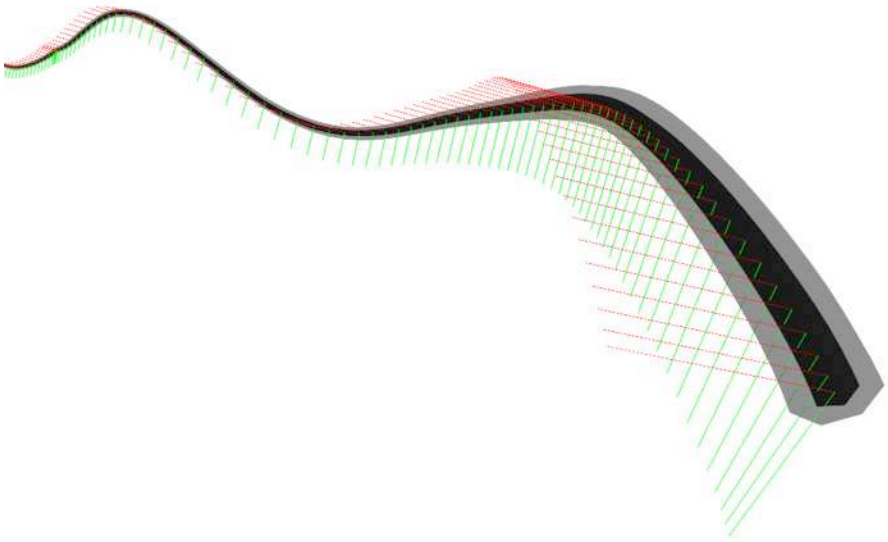


***Figure 27**: Simulated catheter composed on 200 beam elements. The red and green vectors correspond to 2 of the 3 axes of the local reference frame defined at each node. The third axis is tangent to the curve defining the central axis of the catheter.*

To model a wire-like structure, we serially link a series of beam elements (see Figure 27). As a result, for the entire structure the global stiffness matrix $\mathbf{K}$ is computed by summing the contributions of each element, thus leading to the following equilibrium equation

$$\mathbf{K}\mathbf{u} = \mathbf{f} \quad (45)$$

where $\mathbf{K}$ is a band matrix due to the serial structure of the model (one node is only shared by one or two elements) and $\mathbf{u}$ represents a vector of displacements resulting from external forces $\mathbf{f}$. Using directly such a linear model would not represent correctly the geometric non-linearities that a typical wire-like object exhibits. Therefore, we proposed to update Λ for each beam element, and at every time step, by using the solution obtained at the previous time step. The new local stiffness matrices are then assembled in $\mathbf{K}$. This is different from other approaches such as the co-rotational method (Felippa, 2000) or other techniques that remove the rigid body transformation from a given configuration to remain in the linear

domain (Muellller *et al.*, 2004). Here, we do not use the initial configuration as the reference state, but instead use the previously computed solution. By controlling when each new $\mathbf{K}$ is going to be computed, we can ensure we remain in the linear domain for each incremental step, leading to a correct, global deformation. When using this approach, however, the model could exhibit a inelastic behavior, i.e. in the absence of forces or torques, the model would only return to the previous state, not the reference configuration. We overcome this problem by computing an elastic force $\mathbf{f}_e$ on each beam, defined as

$$\mathbf{f}_e = -\gamma \mathbf{k}_e \Lambda^T \mathbf{u}_e \quad (46)$$

where γ is a scalar such that $0 < \gamma \leq 1$. This force is added to the external forces $\mathbf{f}$ in the global coordinate system before solving the linear system, and it can be shown that it acts as a damping force, where γ relates to the damping coefficient of the model. To simulate accurately a device such as a guidewire or catheter we use a number of beam elements ranging from 100 to 200 (Figure 27). With 6 degrees of freedom per node, we need to solve a linear system with about 1,000 unknowns at every time step. Although it can be done using iterative methods, it quickly becomes a limitation when integrating constraint due to contacts with the anatomy. To improve the computational performance of the method we proposes the following optimization.

Substructure analysis

As described above, by using an incremental approach the wire-like model is described by a linear relation between forces and displacements which are mofified at every time step. To optimize computational speed, we propose an assembling method of inverted mechanics which makes it unnecessary to assemble the mechanical properties of the devices in $\mathbf{K}$, therefore precluding the need to compute the complete inversion of $\mathbf{K}$. The method we propose is called structural analysis but is very similar in its concept to the condensation method we introduced in (Bro-Nielsen *et al.*, 1996).

Without loss of generality, let us assume that the nodes of the wire-like model have been ordered with the boundary (b) nodes first, followed by internal (i) nodes. Using this ordering we can write the linear system describing the model as a block matrix system

$$\begin{bmatrix} \mathbf{K}_{bb} & \mathbf{K}_{bi} \\ \mathbf{K}_{ib} & \mathbf{K}_{ii} \end{bmatrix} \begin{bmatrix} \mathbf{u}_b \\ \mathbf{u}_i \end{bmatrix} = \begin{bmatrix} \mathbf{f}_b \\ \mathbf{f}_i \end{bmatrix} \quad (47)$$

One can solve the system this way

$$\begin{bmatrix} \mathbf{u}_b \\ \mathbf{u}_i \end{bmatrix} = \begin{bmatrix} \overbrace{(\mathbf{K}_{bb} - \mathbf{K}_{bi}\mathbf{K}_{ii}^{-1}\mathbf{K}_{ib})^{-1}}^{\mathbf{K}_b^{-1}} & -\mathbf{K}_b^{-1}\overbrace{\mathbf{K}_{bi}\mathbf{K}_{ii}^{-1}}^{-\mathbf{H}_{bi}} \\ \underbrace{-\mathbf{K}_{ii}^{-1}\mathbf{K}_{ib}}_{\mathbf{H}_{ib}}\mathbf{K}_b^{-1} & \mathbf{K}_{ii}^{-1} + \mathbf{H}_{ib}\mathbf{K}_b^{-1}\mathbf{H}_{bi} \end{bmatrix} \begin{bmatrix} \mathbf{f}_b \\ \mathbf{f}_i \end{bmatrix} \quad (48)$$

We then use the inverted system in a different way than in (Bro-Nielsen *et al.*, 1996). We decompose the resolution of the entire system in three steps:

- $\mathbf{u}_i^{(1)} = \mathbf{K}_{ii}^{-1}\mathbf{f}_i$ and $\mathbf{f}_b^{(1)} = \mathbf{H}_{bi}\mathbf{f}_i$: this step is named *boundary fixation*, because it computes the internal motion and the subsequent reaction on the boundary as if the motion of boundary was frozen.
- $\mathbf{u}_b = \mathbf{K}_b^{-1}(\mathbf{f}_b + \mathbf{f}_b^{(1)})$ and $\mathbf{u}_i^{(2)} = \mathbf{H}_{ib}\mathbf{u}_b$: this step is named *boundary relaxation* because it computes the boundary motion due to forces on boundary and reaction forces computed during the first step. Internal displacement due to the motion of boundary is also computed.
- $\mathbf{u}_i = \mathbf{u}_i^{(1)} + \mathbf{u}_i^{(2)}$ and $(\mathbf{K}^{-1})_{ii} = \mathbf{K}_{ii}^{-1} + \mathbf{H}_{ib}\mathbf{K}_b^{-1}\mathbf{H}_{bi}$: this step represents the *flexibility assembling* as it sums the displacements of internal nodes computed during steps 1 and 2.

In the case of a serially-linked structure, as the matrix $\mathbf{K}$ is a band matrix, we can use this substructure decomposition in an accumulative way, i.e. by sequentially applying the substructure analysis equations to all the nodes, from the base to the tip (forward), and then from the tip to the base (backward). By first processing the nodes in the forward direction, and using boundary fixation, we accumulate the reaction forces on the boundaries. By processing then the nodes in the backward direction, and using boundary relaxation, we accumulate the displacements.

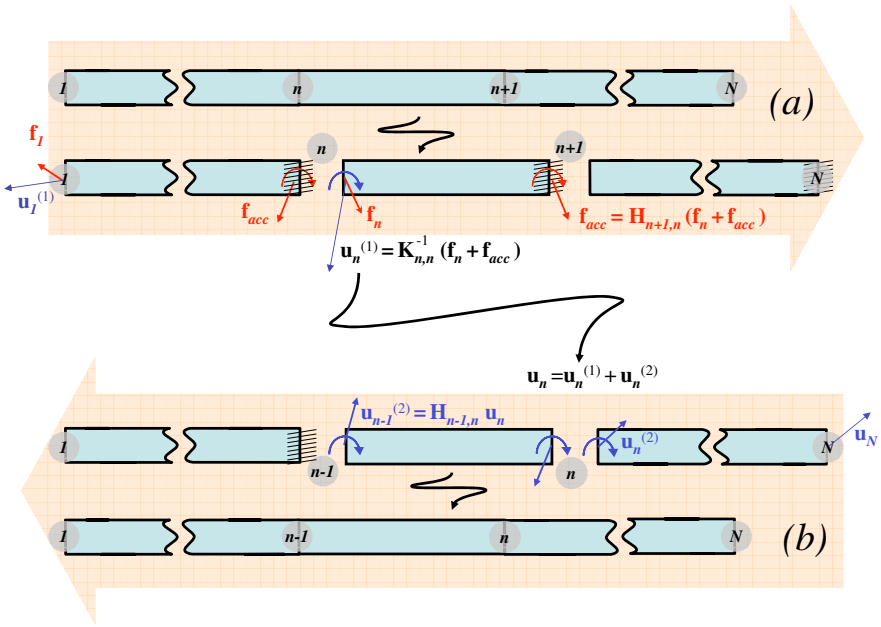


***Figure 28**: Setting boundary conditions: the object is split into a series of substructures. (a) Local displacements $u_n^{(1)}$ and forces are computed after constraining the boundary node of each substructure. (b) Relaxing boundary conditions: correction displacements $u_n^{(2)}$ are applied recursively, starting from node N, at each internal node of each substructure.*

The substructure analysis of the flexible device presents several benefits: a fast update of the whole structure compatible with interactive simulation, and also a mean to compute a the value of the compliance of the mechanical system for a particular node. This information is key in processing contact response during collisions with the anatomy (see Chapter III).

Input: $\mathbf{K}$, f
Output: $\mathbf{u}$ $(= \mathbf{K}^{-1} f)$
$\mathbf{f}_{acc} = 0$;
for $n = 1 \dots N-1$ **do**
  $\mathbf{u}_n^{(1)} = (\mathbf{K}_{n,n})^{-1}(\mathbf{f}_n + \mathbf{f}_{acc})$;
  $\mathbf{f}_{acc} \mathrel{+}= \mathbf{K}_{n+1,n}\, \mathbf{u}_n^{(1)}$
  $\mathbf{K}_{n+1,n+1} \mathrel{-}= \mathbf{K}_{n+1,n}\, (\mathbf{K}_{n,n})^{-1}\, \mathbf{K}_{n,n+1}$
end
$\mathbf{u}_N = (\mathbf{K}_{N,N})^{-1}(\mathbf{f}_N + \mathbf{f}_{acc})$;
for $n = N-1 \dots 1$ **do**
  $\mathbf{u}_n^{(2)} = (\mathbf{K}_{n,n})^{-1}\, \mathbf{K}_{n,n+1}\, \mathbf{u}_{n+1}$;
  $\mathbf{u}_n = \mathbf{u}_n^{(1)} + \mathbf{u}_n^{(2)}$;
end

Solving the system with substructure analysis

Input: $\mathbf{K}$
Output: $[(\mathbf{K}^{-1})_{1,1} \dots (\mathbf{K}^{-1})_{N,N}]$,
$[\mathbf{H}_{1,2} \dots \mathbf{H}_{N,N-1}]$,
$[\mathbf{H}_{2,1} \dots \mathbf{H}_{N-1,N}]$
for $n = 1 \dots N-1$ **do**
  $(\mathbf{K}^{-1})_{n,n} = (\mathbf{K}_{n,n})^{-1}$;
  $\mathbf{K}_{n+1,n+1} \mathrel{-}= \mathbf{K}_{n+1,n}\, (\mathbf{K}_{n,n})^{-1}\, \mathbf{K}_{n,n+1}$:
  $\mathbf{H}_{n+1,n} = \mathbf{K}_{n+1,n}\, (\mathbf{K}_{n,n})^{-1}$;
end
$(\mathbf{K}^{-1})_{N,N} = (\mathbf{K}_{N,N})^{-1}$;
for $n = N-1 \dots 1$ **do**
  $\mathbf{H}_{n,n+1} = (\mathbf{K}_{n,n})^{-1}\, \mathbf{K}_{n,n+1}$;
  $(\mathbf{K}^{-1})_{n,n} \mathrel{+}= \mathbf{H}_{n,n+1}\, (\mathbf{K}^{-1})_{n+1,n+1}\, \mathbf{H}_{n+1,n}$
end

Computing compliance using substructure analysis

***Table 2**: Algorithms describing the two main aspects of the substructure analysis: how to solve the system describing the complete wire structure, and how to compute the compliance of the whole mechanical system.*

Dynamic model

Following this work on catheter and guidewire modeling for interventional radiology simulation, we proposed a second approach, initially aimed at the description of embolization coils. However, as we will see, the proposed model has applications beyond coils and even beyond interventional radiology.

Different types of coils can be used for embolization. Most of them have a core made of platinum, and are sometimes coated with another material or a bio-logically active agent. All types are made of soft platinum wire of less than a millimeter diameter and therefore are very soft. The softness of the platinum allows the coil to conform to the often irregular shape of the aneurysm, while the diameter, length and shape of the coil are chosen based on the shape and volume of the aneurysm, as well as the size of the neck of the aneurysm. In most cases, several coils are required to completely fill the aneurysm and maximize the chances to clot. The proposed coil model uses a series of serially linked beam elements, similarly as proposed in (Cotin *et al.*, 2005) for simulating catheters and guidewires. However, we introduce several modifications to take into account for the particular nature of coils. A different approach for solving the mechanical system of connected beams is also proposed.

To model the deformation of the coil, we still rely on three-dimensional beam theory but since coils exhibit a more important dynamic behavior during their deployment than catheter or guidewires during navigation, we proposed a dynamic formulation of the model. Starting from (Eq. 13) we can write

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{K}(\mathbf{u})d\mathbf{u} = \mathbf{f}(\mathbf{u}) - \overline{\mathbf{f}} \quad (49)$$

where $\mathbf{u}=(\mathbf{x}-\mathbf{x}_0)$ is the vector of displacement of the nodes, $\mathbf{K}(\mathbf{u})$ is a band matrix due to the serial structure of the model, $\mathbf{f}$ represents the external forces applied to the coil, and $\overline{\mathbf{f}}$ corresponds to the elastic energy / force accumulated in the structure. Assuming lumped masses at the nodes, the mass matrix $\mathbf{M}$ is a diagonal matrix and the damping matrix $\mathbf{D}$ is defined as a linear combination of the stiffness and mass matrices $\mathbf{D} = \alpha\mathbf{M}+\beta\mathbf{K}$, known as Raleigh damping. Note that each node is described by six degrees of freedom, three of which correspond to the spatial position, and three to the angular position of the node in a global reference frame.

The elementary stiffness matrix $\mathbf{k}_e$, introduced above, is a 12×12 symmetric matrix that relates spatial and angular positions of each end of a beam element to the forces and torques applied to them (see eq. 43). Each beam stiffness matrix is initially calculated in local coordinates, defined by a reference frame associated to the first node of the beam. In this reference frame, only deformations (bending, torsion, elongation) are measured. As in the previous method, for the entire structure describing a catheter or guidewire, the global stiffness matrix $\mathbf{K(u)}$ needs to be recomputed at every time step. This is done by summing the contributions of each beam element, through its elementary stiffness matrix $\mathbf{K_e}$. In fact, only the transformation matrix $\mathbf{\Lambda}$ introduced in (Eq. 44) needs to be recomputed at every time step, while $\mathbf{k_e}$ remains constant. As long as the discretization of the flexible device is fine enough, the deformation of each beam in its local frame will remain small, and $\mathbf{k_e}$ can be considered constant. Boundary conditions are specified by defining a particular translation or rotation for the first node of the model to represent user control of the device (the coil is manipulated by pushing and twisting a wire). Since the first node of the model is constrained, the first beam equation is used to update the local frame for the second node, thus allowing the second beam to be computed in a reference frame where no rigid transformation occurs. By repeating this process through the whole structure, we can compute $\mathbf{\Lambda}$ for each beam element and therefore determine $\mathbf{K_e}$. This method is closer to the co-rotational approach (Felippa, 2000) than the incremental approach proposed in (Cotin *et al.*, 2005) and permits to model the geometric non-linearities that occur during the deformation of the coil.

Equation (Eq. 49) is integrated in time using a implicit integration scheme (Euler implicit) and then solved using an optimized linear solver that takes advantage of the nature of our model. All beam elements being serially connected, the resulting stiffness matrix K is a tri-diagonal matrix with a band size of 12 (since each $\mathbf{K_e}$ is a 12×12 matrix). Since the mass and damping matrices are also diagonal, we solve the linear system using the algorithm proposed by Kumar *et al.* (Kumar *et al.*, 1993). The solution can be obtained in $O(n)$ operations instead of $O(n^3)$. This allows computation times of less than 10 ms for a coil composed of 100 beam elements on a computer with a Core2Duo processor running at 2.66GHz. In addition, qualitative and quantitative validations have been performed in order to verify the behavior of our coil modeling. Results show an excellent adequacy between the real-time model and experimental data, with differences ranging from 9.8% to 4.17% between the simulated and real coil.

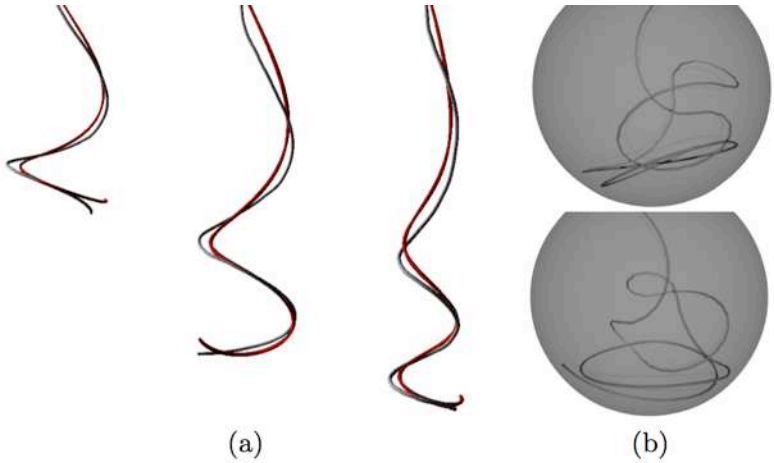


***Figure 29**: (a) Visual comparison of our coil simulation (in red) with the reconstructed coil model from our experiments (in yellow) at different stages of the deployment. (b) Simulation of the deployment of a bird-cage coil in a simple virtual aneurysm using the same parameters as for the helical coil but a different rest shape.*

Visualization devices

Visualization holds a key place in Medicine. It is undoubtedly the main feedback during a medical procedure and one of the most frequent means of evaluating a pathology. Techniques such as X-ray images, fluoroscopy, computed tomography, mammography, magnetic resonance imaging, or ultrasound imaging are now frequently used pre-operatively, and some of these techniques, such as magnetic resonance imaging, fluoroscopy or ultrasound imaging are used intra-operatively. From these techniques were also derived therapeutic techniques such as radiotherapy or high-intensity focused ultrasound therapy.

While medical simulation is not concerned with imaging modalities that are only used for diagnostic purpose, we need to be able to reproduce the different types of visualization possible during a procedure. This mainly involves visible light, fluoroscopic, and ultrasound imaging. Visible light imaging is the most direct rendering technique since all rendering libraries such as *OpenGL* generate images in the visible spectrum. However, actual images generated by laparoscopic cameras or ophthalmology microscopes present particular optical characteristics that need to be simulated to produce realistic images. Also, the great level of detail that these visualization techniques provide requires the creation of fine anatomical models and detailed textures to achieve a similar level of realism. For other visualization techniques such as ultrasound or fluoroscopy, we need to model the actual physical process of the imaging device. This is more challenging since this process is often complex and graphics processing units are not designed for this type of rendering.

In this section we describe various techniques for rendering a simulated angiography with a high level of fidelity. While prior work as well as commercial products rely on polygon-based rendering techniques to simulate X-ray images, we propose a method based on volume rendering. Such a technique can provide a high level of realism yet can be optimized for real-time rendering on current consumer GPU.

Simulated fluoroscopy

In this section we present the work we did with M. Manivannan and V. Pegoraro during, respectively, a postdoctoral fellowship and an summer internship. This work was part of the development of a training system for interventional radiology (see Chapter V).

While most simulation systems have been developed in the context of laparoscopic surgery and therefore require to provide visual feedback in the visible spectrum, interventional radiology is a medical discipline that replies extensively on intra-operative X-ray images. This requires the development of fast and accurate methods of simulating X-ray images. There are three different classes of algorithms that have been developed in the context of interventional radiology training: actual fluoroscopic images, polygon-based techniques, and volume rendering techniques. While the use of actual fluoroscopic images limits the interaction to a fixed viewpoint, polygon models require that every anatomical structure to be rendered is previously segmented, thus limiting the level of detail that can be rendered. Therefore it seems that using a method relying on volume rendering, combined with a high-resolution CT scan volume of the anatomical structure of interest, would be an ideal approach.

There are many possible ways to implement 3D volume rendering depending on the goal of the application, but very few actually address the problem of rendering X-ray images from CT scan volumes at real-time frame rates. A common approach to the problem of X-ray rendering is to process CT volumetric images (Mullick *et al.*, 1996) (Park *et al.*, 1996) (van Walsum *et al.*, 1997) since CT images are actually created by combining a radial sequence of X-ray images. One approach consists in applying

almost the inverse process of what is used in Computed Tomography by computing Fast Fourier Transforms to extract the frequency spectrum of the projection of the data set in the Fourier domain and then computing the projection by applying an inverse 2D Fourier transformation to that spectrum (Napel *et al.*, 1991). By reducing the computational complexity from 3D to 2D this allows for faster rendering although not real-time. Another approach is to use a volume rendering technique. Volume rendering is based on light interaction with the voxels in a volume of data. In the Ray Casting approach (Park *et al.*, 1996), for each ray cast through the data set, the stepping and interpolation stages generate and accumulate a number of sample points along that ray. By specifying how the intensities of voxels traversed by a ray are combined with each other, it is possible to generate images similar to X-ray images. A third way of dealing with the problem consists in applying similar principles, but instead of dealing with very large volumes of voxels, the X-ray image is generated from polygonal models (Cotin *et al.*, 2000c). Although it allows the creation of rather realistic images in real-time, this technique requires segmenting the anatomy, defining an attenuation coefficient for each structure, and evaluating the length of each polygonal volume that is penetrated by the X-ray beam, thus making it less flexible and accurate than volume rendering techniques. Our approach uses hardware implementation of display traversal in graphics boards to accelerate volume rendering.

The basic principle upon which X-ray imaging methods depend is the attenuation of the X-ray beam as it passes through tissues of different densities. The X-ray beam is produced in an X-ray tube and exits from the tube where it is collimated, then enters the patient in the area of interest, and finally is partially absorbed by the patient's tissues. The portion of the beam that penetrates the patient reaches a cassette that contains screens. The X-rays energize crystals in the screens, which produce light proportional to the amount of X-ray energy that energized them. The light generated by the screens exposes radiographic film, which is processed to produce an image. Following the same principles, fluoroscopy is an X-ray test that uses a continuous beam of X-rays to follow movement in the body. During fluoroscopy, X-rays are directed continuously at an area of the body, and the resulting pictures are displayed on a monitor. Fluoroscopy can be used to evaluate lung or heart motion but is mostly used to guide the placement of medical devices in the body, such as during angiography. A contrast agent that shows up on X-rays can be injected into a blood vessel during fluoroscopy to make the outline of blood vessels visible. It is in part because of the continuous exposure of the patient to X-ray radiation that a high level of training could help minimize exposure time. In order to simulate accurately the fluoroscopic process, we must first generate high quality X-ray images and second generate these images at a frame rate of about 20 images per second. Since photon energy, atomic number, density, electron density, and thickness of the material affect absorption and scattering of the photon (Napel *et al.*, 1991), our approach and main contribution will be to attempt at simulating as many characteristics as possible, in real-time, to produce highly realistic images.

Fluoroscopy is an imaging modality that uses a continuous beam of X-rays to follow movement in the body. It follows the same principle as X-ray imaging, where the attenuation of a X-ray beam depends on the density and thickness of the tissue it traverses. The attenuation of an X-ray beam traversing a thin slice of homogeneous material is given as

$$I = I_0 e^{-\mu d} \qquad (50)$$

with I_0 the initial X-ray beam intensity, μ the coefficient of linear attenuation of the material, and d the traversed material thickness. The attenuation coefficient varies with the cube of the atomic number of the material being traversed. Therefore, mass attenuation coefficient of bone (atomic number about 11.6) is significantly higher than the fat (atomic number 7.4) explaining why bone and fat appear highly contrasted

in an X-ray image. When the beam traverses several structures of various thickness and attenuation coefficients, the previous continuous equation can be rewritten as

$$I = I_0 e^{-\sum_i \mu_i d_i} \quad (51)$$

where μ_i the linear attenuation coefficient of structure i, and d_i the thickness of the structure. One can also notice that this equation can be used to describe the change in intensity of a beam that would traverse a discretized representation of the anatomy (such as a CT scan for instance) where d_i would correspond to the slice thickness along the ray and μ_i would be the attenuation coefficient of the anatomic structure sampled in slice *i*.

To simulate accurately the fluoroscopic process, we have developed a volume rendering approach which renders a CT scan data set using a particular transfer function.

Equation (Eq. 50) describes the decrease in intensity when passing through an object. If we consider a slice of the CT scan as a thin cross-section of the same object, then equation (Eq. 50) can be approximated by

$$\Delta I = -\mu I \Delta d \quad (52)$$

Assuming Δd is known and constant, we can write

$$\Delta I = I_d - I_s = -\alpha I_s \quad \text{and} \quad I_d = I_s(1-\alpha) \quad (53)$$

Such a function can be implemented very efficiently using OpenGL with glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA) for back-to-front X-ray renderingThe resulting fragment is then blended into the frame buffer using the blending functions such as `glBlendFunc(GL_ONE, GL_SRC_ALPHA)`.

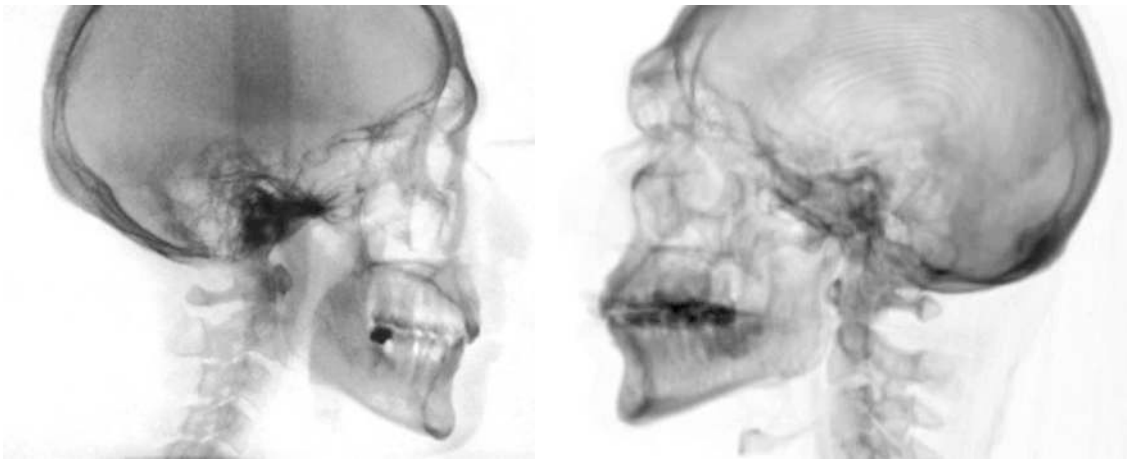


***Figure 31**: Simulated and actual X-ray images. The real-time rendered simulated X-ray image is on the right, while the real X-ray image is on the left.*

Multi-scale particle-based angiography simulation

Simulating X-ray or fluoroscopic images is an essential step for training interventional radiologists. Yet, to recreate an angiogram (i.e. an imaging test that uses X-rays and a contrast agent to visualize blood vessels) we also need to simulate how the presence of a contrast agent alters the fluoroscopic images.

Our approach to simulating angiographic images relies on similar principles as for X-ray images. We use volume rendering techniques to achieve a high level of realism in the resulting image. Also, since the X-ray image is rendered using the same approach, it guarantees the rendering of the anatomy and the rendering of the vascular structures will blend perfectly. This work was published in 2007 in the proceedings of MICCAI conference (Wu *et al.*, 2007).

Particles are rendered as point sprites which are squares to which we apply a circular texture mapping. To visualize the propagation of contrast agent we create a volumetric representation of the vessels lumen by "filling" the vessels with particles. These "spherical voxels" are associated with a particular location x along each vessel, and the intensity and transparency properties of the particle are a function of the contrast agent concentration at this location $C(x,t)$. Such particles can be rendered efficiently on recent GPUs. However, to deal with very detailed vascular models where the vessels radii vary from 0.5 mm to 17.5 mm, using equally sized three-dimensional particles to represent the vessel's lumen can lead to a very large number of particles (more than 40 millions in our case) which can no longer be rendered in real-time.

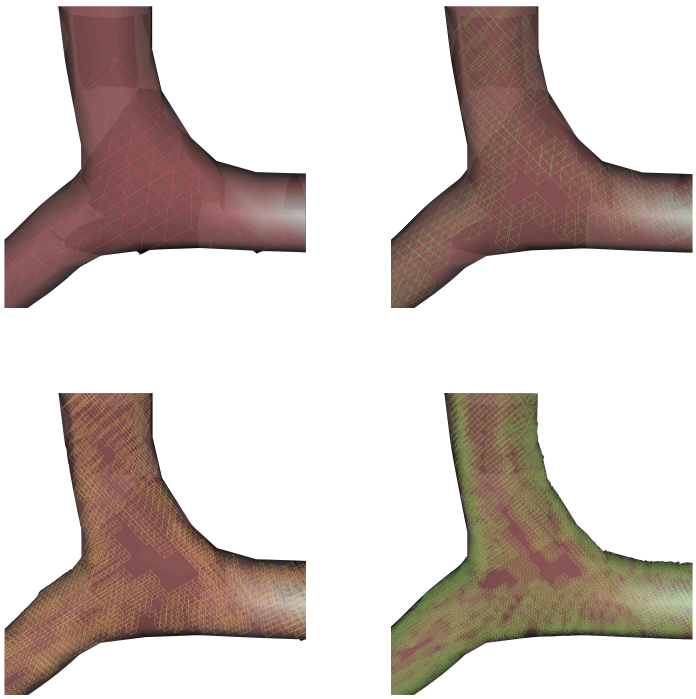


***Figure 32**: Different levels of partitioning of the lumen of a bifurcation between vessels. Smaller cells are create near the boundaries of the vessel to discretize more precisely the volume of the vessel. Larger cells are used near the center to reduce the overall number of cells. These cells are used to create point sprites for rendering contrast agent.*

To maintain both high-resolution visualization and real-time rendering, we developed a multi-scale approach based on subdivision. Similarly to a marching cube algorithm, we start from a prescribed initial grid size, and we determine for each voxel if it is located inside, outside or on the boundary of a vessel. External voxels are removed, internal ones are stored, and boundary voxels are subdivided into 8 equally sized sub-voxels. Each sub-voxel is examined against the set of surface polygons intersected by the parent voxel. If a sub-voxel does not intersect with surface and it is inside the surface, then that sub-voxel is labelled as internal and stored. Boundary sub-voxels are again subdivided. The iterative step

continues until the predefined number of subdivisions is reached. As a result, starting from a 2×2×2 mm^3 grid, and using 4 subdivision steps, only 9.2 million multi-scale voxels are generated (and the smallest particle is of size 0.25×0.25×0.25 mm^3).

Different sized particles have different amount of radiation attenuation under fluoroscopy. This is achieved by adjusting each particle's rendering size linearly and its intensity exponentially according to its dimensions. This approach is implemented using a programmable shader and runs at interactive frame rates. Results are very realistic, as illustrated in Figure 33.

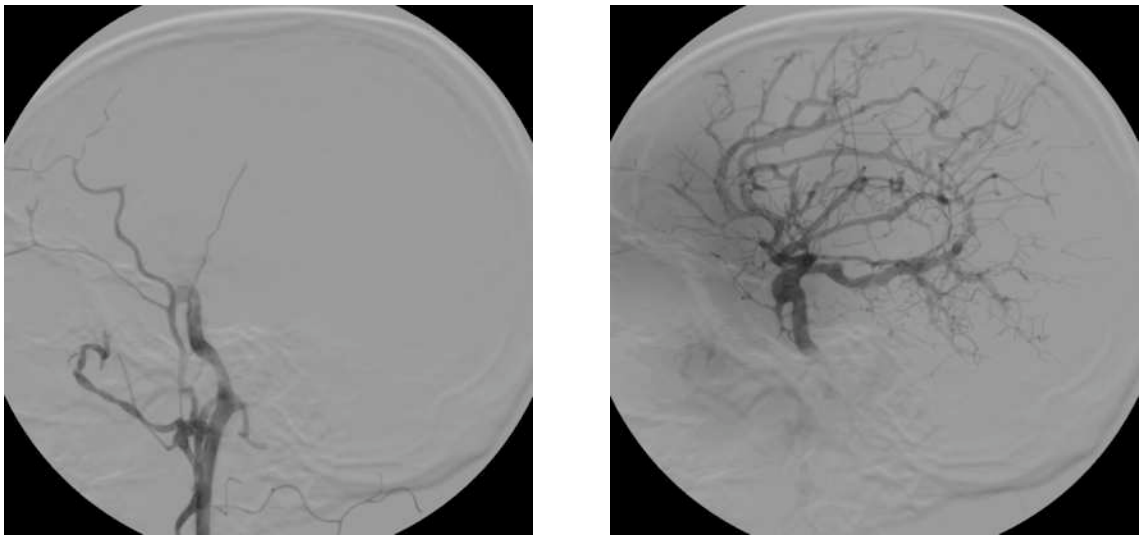


***Figure 33**: An angiography seen at two different times (the image on the right was taken about 1 second after the image on the left). These images are simulated using a contrast agent propagation algorithm and volumetric rendering of the blood vessels. Grey level intensity is proportional to the concentration of contrast agent in the vessels.*

Future directions

In the field of interventional radiology, we have demonstrated that it is possible to reach a very high level of visual realism. Often disregarded as not being a key feature of a simulator, realistic rendering is an important element to reach the state of what is often called "suspension of disbelief" i.e. when the user forgets he / she is dealing with a simulator. Recreating the operating field with as many details as possible, as well as reproducing visual effects such as bleeding, smoke, lens deformation, are all important elements of visualization. Several of these effects rely on advanced models. For instance, bleeding is a visual effect that has been rather poorly simulated in medical simulation. Although computationally expensive, several authors have demonstrated the feasibility of physics-based three-dimensional fluid flow simulation in real-time. Combining visually realistic bleeding with fluid models that can interact with the overall physiology of the simulated patient could lead to very realistic simulations.

Regarding the simulation of medical devices, our work on wire-like structures could easily be expanded to model flexible endoscopes and flexible laparoscopic instruments. With the recent advent of Natural Orifice Transluminal Endoscopic Surgery, this would benefit the development of a new type of simulators.

Summary of contributions

- **Research articles**
  - X. Wu, J. Allard, and S. Cotin. “Real-time Modeling of Vascular Flow for Angiography Simulation”. Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), Volume 4792, pp. 850-857, 2007.
  - J. Rabinov, S. Cotin, J. Allard, J. Dequidt, J. Lenoir, V. Luboz , P. Neumann, X. Wu, and S. Dawson. “EVE: Computer Based Endovascular Training System for Neuroradiolgy”. ASNR 45th Annual Meeting & NER Foundation Symposium, pp. 147-150, 2007.
  - C. Duriez, S. Cotin and J. Lenoir. “New Approaches to Catheter Navigation for Interventional Radiology Simulation”. In: Computer Aided Surgery, vol. 11, p. 300-308, 2006.
  - S. Cotin, C. Duriez, J. Lenoir, P. Neumann, S. Dawson. “New approaches to catheter navigation for interventional radiology simulation”. Proceedings of the MICCAI Conference, MICCAI 2005, pp. 534-542, 2005.
  - X. Wu, V. Pegoraro, V. Luboz, P. Neumann, R. Bardsley, S. Dawson, and S. Cotin. “New Approaches to Computer-based Interventional Neuroradiology Training”. Proceedings of 13th Annual Meeting, Medicine Meets Virtual Reality; January 2005.
  - M. Manivannan, S. Cotin, M. Srinivasan, S. Dawson. “Real-Time PC based X-ray Simulation for Interventional Radiology Training”. Proceedings of 11th Annual Meeting, Medicine Meets Virtual Reality, pp. 233-239, 2003.
- **Patents**
  - Cotin S, Wu X, Neumann P, Dawson S; "Methods and Apparatus for Simulation of Endovascular and Endoluminal Procedures", United States Patent Application, 60/600,188 - PCT application number PCT/US2005/028594, 2005.
- **Software**
  - **EVE: interventional radiology training system**: the algorithm for real-time flexible device models has been integrated in the EVE system, along with the volumetric technique for simulating X-ray images and for rendering angiograms.

Chapter III — Interaction

1. Introduction

Interactivity is a fundamental notion is medical training. Although extensive “theoretical” knowledge in areas such as anatomy and physiology is mandatory, the final and probably most important step in the curriculum is the residency. This is when the theory has to be applied and translated into actions. Each action from the surgeon or interventional radiologist will influence the end result of the procedure. But this interaction with the patient and the anatomy will also guide the physician, help him/her adjust the pre-operative strategy, and eventually become more confident.

To reproduce the complexity of the interactions between the physician, the instruments, and the anatomy, many steps are required. We have seen in the previous chapter that the anatomy and the instruments need to be modeled first, and as accurately as possible. The next step requires to detect potential contacts between the virtual instruments and the virtual anatomy, or between different organs, and to provide appropriate feedback. I In a large number of medical procedures this feedback is essentially visual (deformation of the anatomical structures, bleeding, ...) but in some instances, a haptic or tactile feedback needs also to be provided. This chapter covers essentially the problem of contact modeling, an area where we have had important contributions. An overview of techniques for collision detection with deformable structures is also presented, and haptic feedback is briefly mentioned.

2. Collision detection

Collision detection is a vast problem, well-studied in computer graphics, and, to a lesser extent, in medical simulation. The main challenge with collision detection relates to simulations involving deformable models (although not all simulations in medicine involve soft tissues or flexible devices). When compared to collision detection approaches for rigid bodies, which has been the primary focus of a large number of collision detection methods, there are various aspects that complicate the problem for deformable objects. **Collisions and Self-collisions**: in order to realistically simulate interactions between deformable objects, all contact points including those due to self-collisions have to be considered. This is in contrast to rigid body collision detection, where self-collisions are commonly neglected. Depending on the applications, rigid body approaches can further be accelerated by only detecting one contact point. **Pre-processing**: efficient collision detection algorithms are accelerated by using spatial data structures including bounding-volume hierarchies, distance fields, or alternative ways of spatial partitioning. Such object representations are commonly built in a pre-processing stage and perform very well for rigid objects. However, in the case of deforming objects these pre-processed data structures have to be updated frequently. Therefore, pre-processed data structures are less efficient for deforming objects and their practicality has to be examined very carefully. **Collision Information**: collision detection algorithms for deformable objects have to consider that a realistic collision response requires appropriate information. Therefore, it is not sufficient to just detect the interference of objects. Instead, precise information such as penetration depth of objects is desired. **Performance**: in interactive applications,

such as medical simulation or interactive computer graphics, the efficiency of collision detection algorithms is particularly important.

Collision detection techniques, suitable for simulations involving deformable models, are numerous and it is virtually impossible to cover them all. Among the most relevant work we can mention Quick-Cullide (Govindaraju *et al.*, 2005), an algorithm based on the GPU. The principle consists in rendering objects, in a particular order and along predefined view points, to determine the ones that are potentially in contact. This visibility query is done at object and sub-object levels and is followed by an exact collision detection. This approach allows real-time collision detection even with detailed triangulated objects. In this approach as in many others, collision detection is accelerated by decomposing the object into a hierarchy of primitives, the finer one often corresponding to the triangles in the object. Using Bounding Volume Hierarchies have proven to be among the most efficient data structures for collision detection, although they have mostly been applied to rigid body collision detection. The choice of the type of hierarchy, of the number of levels, or update strategy is mostly what differentiate one approach from another. For instance, in (Pai, *et al.*, 2004), a hierarchy of bounding spheres is used but it accounts for the deformation of the object to prevent an update of the hierarchy at every time step. Bounding Volume Hierarchies are also used to accelerate continuous collision detection, i. e., to detect the exact contact of dynamically simulated objects within two successive time steps. Therefore, bounding volumes do not only cover object primitives at a certain time step, they also enclose the volume described by the linear motion of a primitive within two successive time steps (Redon *et al.*, 2002).

Recently, stochastic methods (sometimes referred to as “inexact” methods) have become a focus in collision detection research. This idea is motivated by several observations. First, polygonal models are just an approximation of the true geometry. Second, the perceived quality of most interactive 3D applications does not depend on exact simulation, but rather on real-time response to collisions (Uno *et al.*, 1997). Therefore, it can be tolerated to improve the performance of collision detection, while degrading its precision. Raghupathi *et al.* (Raghupathi *et al.*, 2004) adapted a stochastic technique to detect the collisions occurring in the intestinal region when a surgeon manipulates the small intestine. Both, the self-collisions within the intestine and the collisions with the mesentery were handled in this application. The system achieved real-time performances for a deformable model undergoing a few hundred collisions.

Distance fields specify the minimum distance to a closed surface for all points in the field. The distance may be signed in order to distinguish between inside and outside. Representing a closed surface by a distance field is advantageous because there are no restrictions about topology. Further, the evaluation of distances and normals needed for collision detection and response is extremely fast and independent of the geometric complexity of the object. Distance fields have been employed to detect collisions and even self-collisions in non-interactive applications. They provide a highly robust collision detection, since they divide space strictly into inside and outside. Efficient algorithms for computing distance fields have been proposed recently. Even though distance fields need to be updated in the case of deformable objects, in some instances (e.g. linear deformations) they are fast enough for interactive applications.

Recently, several image-space techniques have been proposed for collision detection (Baciu *et al.*, 2002) (Heidelgerg *et al.*, 2003). These approaches commonly process projections of objects to accelerate collision queries. As such, they are especially appropriate for environments with dynamically deforming objects as they do not require any pre-processing. Furthermore, image-space techniques can commonly be implemented using graphics hardware, which can potentially lead to very efficient computations.

More details about other approaches for collision detection with deformable objects can be found in a survey by Teshner *et al.* (Teschner *et al.*, 2005).

Implicit Surface Modeling

For organic shapes (i.e. shapes that do not exhibit sharp features) a fast collision detection can be performed by using an implicit description of the surface, rather than a triangulation. The surface is then described using a combination of geometrical primitives and a convolution filter. The primitives can be either points, segments, triangles or other simple shapes. The convolution filter h is defined as a function from $\mathbb{R}^3 \rightarrow \mathbb{R}^+$ with a finite support or fast decay to 0. The resulting surface is $f(P) = h(P) \otimes s = I_v$, where I_v is an isosurface value and $f(P)$ the potential at P. Figure 34 illustrates two shapes modeled using different primitives and a gaussian convolution filter. In addition, pathologies such as tumors or aneurysms can be modeled by locally modifying the potential field.

Given a function g defined as $g = f - I_v$ and a point P at two different time steps t and $t+1$, the collision detection consists in finding where $[P_t, P_{t+1}]$ intersects the surface f. This is equivalent to finding the first root i_0 of g on the interval $[P_t, P_{t+1}]$. This is achieved using a modified version of the Newton-Raphson algorithm. From i_0 and $-\nabla g(i_0)$ the surface gradient at i_0, we can compute a linear approximation of the surface. This approximation defines the tangent plane $-\nabla g(i_0) \times P = \|i_0\|$ at i_0 which parameters can be used by a contact response algorithm.

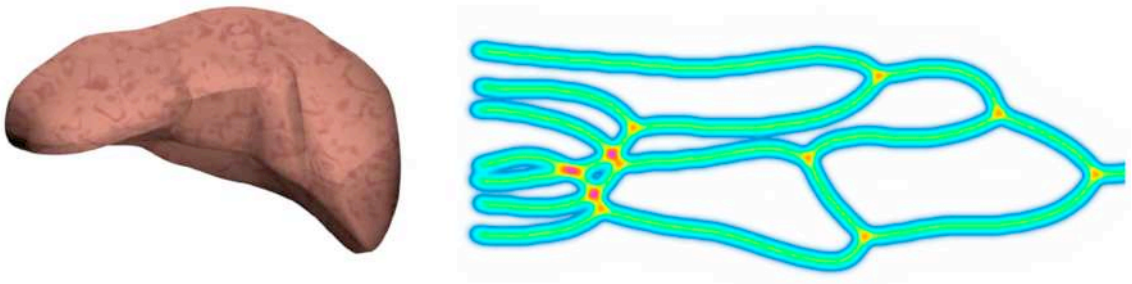


***Figure 34**:* Two examples of convolution surfaces: a liver modeled using a set of points a primitives (left) and a vascular network modeled using a set of segments as primitives. Both use a gaussian function as convolution filter.

3. Collision response

The constant improvements in the field of medical simulation have led to the creation of more and more complex environments, with objects of different nature (rigid, deformable or even fluid) interacting with each other. While a large effort has been put recently toward collision detection between deformable structures, little has been done regarding the precise modeling of contacts between such objects. The majority of the simulations do not explicitly address the problem of modeling the contacts that occur between these different structures. This is particularly true in the case of deformable objects, where contacts are often modeled in a simple way, leading to inaccurate results. This can be explained in part

by the difficulty of developing efficient approaches to this problem that can handle the requirements of interactive simulations, the complexity of the deformation model, as well as the large number of degrees of freedom involved.

Contact modeling is a challenging problem, relevant to many applications of interactive simulation. The way contacts are handled plays a very important role in the overall behavior of the interacting objects. The choice of the contact model, its accuracy, or the inclusion of friction forces, highly influence the post-impact motion of the interacting objects. Additionally, when a contact occurs, immediate changes in the dynamic behavior of the objects occur. Such changes often lead to instabilities or visual inconsistencies in the simulation, which can be critical in medical applications. Finally, when multiple objects are in contact, the solution space for the new, non-interpenetrating, configuration is reduced. The high computational complexity involved in resolving such contacts can become an important bottle neck of the simulation, sometimes more time consuming than the computation of the dynamics or deformation of the objects.

Contact modeling has been extensively studied in Mechanics, and research on modeling the non-smooth dynamic behavior of objects in contact remains an active topic (Acary *et al.*, 2008). In the field of Computer Graphics, several solutions have been proposed to address this problem. The most popular approach is the penalty method which consists in defining a contact force $f^c = k\,\delta$ at each contact point where δ is a measure of the interpenetration between a pair of colliding objects, and k is a stiffness parameter. This stiffness parameter must be large enough to avoid any visible interpenetration, however, its value cannot be determined exactly. Instead, the choice of the value of k depends of the nature of the objects, the type of interactions, and other elements of the simulation, which leads to various heuristics to determine the ideal stiffness parameter. Yet, no matter how k is chosen, interpenetrations between the colliding objects can only be reduced, not entirely suppressed. This a direct consequence of the method itself, which generates forces only when the interpenetration distance δ is negative (assuming δ is chosen to be negative when an interpenetration exists, and positive when the objects are no longer in contact). This is in contradiction with Signorini's law of contact which states that there exists a complementarity relation between the interpenetration distance δ and the normal contact force f^c at the point of contact, i.e. $0 \leq \delta \perp f^c \geq 0$. In addition, if an explicit time integration scheme is used, and k is large, very small time steps are required to guarantee the stability. As explicit integration schemes are conditionally stable, using a penalty method therefore requires that two criteria are met: k must be large enough to limit interpenetrations, and the time step must be small. Overall this makes this approach, initially simple, rather inefficient for handling contacts. A possible improvement over the penalty method can be achieved through the use of an implicit integration scheme. Implicit methods have the advantage of providing more stable simulations even with rather large time steps (Baraff et al., 1998). When combining an implicit integration scheme with a penalty method, it becomes possible to use large stiffness values without compromising the stability of the simulation. However, solving the resulting stiff and non-smooth system can quickly become computationally prohibitive.

Collision responses can also be computed using impulse-based methods. Originally employed to handle contact between rigid object (Mirtich et al., 1995), these methods have been extended to deformable bodies (Bridson *et al.*, 2002). Impulse-based techniques rely on velocity correction, and do not involve constraints nor forces. Whenever two objects are colliding, each one is subject to an opposite impulse which avoid the interpenetration. Hence, a body resting on a table is continuously experiencing collisions with the table, and experience associated impulses. Using these methods, each type of contact, i.e.

colliding, rolling, sliding, and resting, can be simulated in a similar way. However, these methods are usually rather unstable and are inadequate for the simulation of simultaneous contacts.

Overall, the methods described above share an important limitation when dealing with multiple contacts: they consider each contact independently while in reality they are coupled. This limitation can be solved using constraint-based techniques, which can solve "exactly" the contact problem (i.e. no interpenetration at the end of the time step). Such approaches often rely on the use of Lagrange multipliers, which are appropriate for handling bilateral constraints (Galoppo *et al.*, 2006). However, contacts between objects intrinsically define unilateral constraints. Using techniques based on Lagrange multipliers, deformable objects in contact will appear stuck at the end of the time step. Improvements over Lagrange multipliers techniques are possible by using a Linear Complementarity Problem (LCP) formulation deriving from Signorini's law. The solution of the LCP gives an accurate description of the contact forces needed to zero out the interpenetration, and prevents objects from sticking to each other. Pauly *et al.* (Pauly *et al.*, 2004) for instance have proposed such an approach to solve contacts between quasi-rigid objects. The authors use a Lemke solver to compute a contact-free configuration from the LCP formulation. By expanding the LCP, or by using a non-linear solver, the formulation can be extended to model both static and dynamic friction. For rigid objects, see for instance (Anitescu *et al.*, 1999) or (Baraff, 1989), and for deformable objects, see (Pauly *et al.*, 2004) or (Duriez *et al.*, 2006b). Computationally efficient methods for solving linear complementarity problems have been proposed, thus making such approaches appealing even for interactive simulations. Yet, when dealing with deformable models, real-time computation of the solution is almost impossible since the LCP algorithm requires the computation of the inverse of the stiffness matrix for each object in contact. While this inverse can be pre-computed in the case of linear elastic models (Duriez *et al.*, 2006b), this is not possible for non-linear deformable models.

In 2005, we started to study the problem of solving complex contacts in real-time, motivated by our work on interventional radiology simulation (see Chapter V). Whether it relates to catheter navigation in a vessel or coil deployment in an aneurysm, interventional radiology simulation illustrates very clearly the need for accurate collision response. Additionally, the insertion or deployment of flexible devices in very tight spaces can lead to a large number of contacts. When combined with sliding conditions, the computation of the collision response cannot be properly solved using typical methods. Therefore, we have proposed three new approaches to the problem of real-time accurate contact modeling. The first contribution used a substructure analysis as described in Chapter II to improve the computational efficiency of the method (Cotin *et al.*, 2005). The second approach relied on Lagrange multipliers combined with a status method to handle unilateral constraints (Dequidt *et al.*, 2007). Our most recent contribution (Saupin *et al.*, 2008a) relies on the computation of an approximation of the inverse stiffness matrix to minimize computational times in the case of non-linear deformable models.

Signorini's law of contact

It is important to analyze in more details the contact law briefly introduced above. In the case of two objects in contact, for each point of the contact area, Signorini's law states that

$$0 \leq \delta_n \perp f_n^c \geq 0 \quad (54)$$

where δ_n is the interpenetration distance evaluated at the point of contact (shortest euclidian distance to the other object's surface) and $f_n{}^c$ is the amplitude of normal force needed to solve the contact. In the case of frictional sliding, a tangential component $f_t{}^c$ is introduced, leading to a contact force $f^c = f_n{}^c + f_t{}^c$.

From a mathematical stand point, this law translates the orthogonality between the contact force and the interpenetration distance $0 \leq \delta_n \perp f_n^c \geq 0$ meaning that either a interpenetration occurs, requiring a non-null normal force to bring back the contact, or that the constraint is not violated because the distance to the surface is non-null, therefore no force is required to correct the position. In the case of a dynamic problem, a velocity formulation of Signorini's law is often used

$$0 \leq \dot{\delta}_n(t) \perp f_n^c \geq 0 \quad \text{if } \delta_n(t) = 0 \qquad (55)$$

where $\dot{\delta}_n(t)$ describes the relative velocity, along the normal at the point of contact, between the two objects.

Time stepping

In the different contributions described in the remainder of this section, we used a time-stepping scheme, i.e. the time step for discretizing in time is constant and there is no limitation on the number of discontinuity that could happen during a time step (Anitescu *et al.*, 1999). Although this can lead to excessive dissipation if the time step is too large, such a scheme provides stable simulations. The other possible scheme is called event-driven time integration, where irregular time increments are used, based on contact occurrence. If we consider a regular time interval $[t_1, t_2]$ of length $\Delta t = t_2 - t_1$, we use an Euler implicit time integration scheme and approximate (Eq. 13) by

$$(\mathbf{M} + \Delta t\mathbf{B} + \Delta t^2\mathbf{K})\Delta\dot{\mathbf{u}} = -\Delta t^2\mathbf{K}\dot{\mathbf{u}}_1 - \Delta t(\mathbf{f}_1^i + \mathbf{f}_2) + \Delta t\mathbf{r} \qquad (56)$$

where $\mathbf{r}$ are the contact forces, and the nonlinear function $F(\dot{\mathbf{u}},\mathbf{u})$ introduced in equation (Eq. 13) has been approximated using a first order approximation

$$F(\dot{\mathbf{u}} + \Delta\dot{\mathbf{u}}, \mathbf{u} + \Delta\mathbf{u}) \simeq \mathbf{f}_1^i + \mathbf{D}\Delta\dot{\mathbf{u}} + \mathbf{K}\Delta\mathbf{u} \qquad (57)$$

where $\Delta\mathbf{u}=\mathbf{u}_2-\mathbf{u}_1= \Delta t\,\dot{\mathbf{u}}_2$, $\Delta\mathbf{v}=\mathbf{v}_2-\mathbf{v}_1$ and $\mathbf{f}_1^i$ is the vector of internal forces at t_1 and $\mathbf{f}_2$ is the vector of external forces at t_2.

Contact correction

When dealing with numerical simulations, the motion of the objects is discretized into a series of time steps. Some of the external forces are known at the beginning of a time step (gravity, user-specified forces, etc.) while others only appear during the time step, and depend on the current state of the mechanical system. This is the case of the contact forces. Such forces are called implicit, while the known ones are called explicit. However, separating explicit and implicit forces independently is a consequence of the superposition principle, and therefore can only be applied if the equations of motion are linearized.

Assuming the deformable model is linear (or has been linearized during the time step), one way of dealing with both implicit and explicit forces is to split the computation in two steps. First we compute a displacement called free motion, noted $\mathbf{u}_{free}$, in which we take into account only the explicit forces, not the contacts. Second, after after applying the displacement vector $\mathbf{u}_{free}$ to the object, a collision detection is performed to determine if interpenetrations exist between objects. If that is the case, a corrective

motion $\Delta\mathbf{u}$ needs to be computed such that the displacement $\mathbf{u} = \mathbf{u}_{free} + \Delta\mathbf{u}$ verifies the unilateral constraints (i.e. Signorini's law). Knowing the vector of contact forces $\mathbf{f}^c$, we can determine $\Delta\mathbf{u}$ using the following relation

$$\Delta\mathbf{u} = \mathbf{u}_2 - \mathbf{u}_{free} = \left(\frac{1}{\Delta t^2}\mathbf{M} + \frac{1}{\Delta t}\mathbf{B} + \mathbf{K}\right)^{-1} \mathbf{f}^c \quad (58)$$

In addition to this equation, initial and boundary conditions are classically added to the dynamic problem. We can then define the compliance matrix as

$$\mathbf{C} = \left(\frac{1}{\Delta t^2}\mathbf{M} + \frac{1}{\Delta t}\mathbf{B} + \mathbf{K}\right)^{-1} \quad (59)$$

which reduces to $\mathbf{C} = \mathbf{K}^{-1}$ in the static case.

Delassus operator

Once a mean of computing the contact correction for a particular location has been determined, the next step consists in defining how multiple contacts can be solved simultaneously. This is an important characteristic of the various approaches described below: in each of our methods, we guarantee a contact-free configuration at the end of the time step. This is essential in the case of highly constrained configurations (such as a catheter navigating through tortuous vessels) as it prevents visually or mechanically inconsistent configurations, thus making the simulation more realistic.

The mechanical coupling between the various contact points can be represented by the Delassus operator $\mathbf{W}$ (Moreau *et al.*, 1996) defined as

$$\mathbf{W} = \mathbf{H}\mathbf{C}\mathbf{H}^T \quad (60)$$

where the operator $\mathbf{H}$ is a block matrix in which each block is a transformation matrix from the local contact space to the global reference frame where the equations of motion are defined. The contact space for a given contact point is defined by a reference frame composed of a normal vector and two tangent vectors to the contact surface. The relative displacement $\Delta\boldsymbol{\delta}$ and contact forces $\mathbf{f}^c$ between all contact points can then be defined

$$\Delta\delta = \delta - \delta_{free} = \mathbf{H}\Delta\mathbf{u} \qquad \mathbf{f}^c = \mathbf{H}^T\mathbf{r} \quad (61)$$

thus leading to

$$\delta = \mathbf{H}\mathbf{C}\mathbf{H}^T\mathbf{r} + \delta_{free} \quad (62)$$

which can be written using the Delassus operator as

$$\delta = \mathbf{W}\mathbf{r} + \delta_{free} \quad (63)$$

Equations (Eq. 59) and (Eq. 54) define a Linear Complementarity Problem (LCP) which can be solved using different algorithms. To achieve real-time performances, the computation of the LCP must be very

efficient. As it requires the computation of the matrix $\mathbf{W}$ which depends on $\mathbf{C}$, optimization strategies need to be developed to compute $\mathbf{C}$ and the LCP very efficiently, even for non linear models or in the case of a large number of contacts.

Once the contact problem is solved, we obtain the vector of contact force $\mathbf{r}$ which is defined in the contact space. It has to be transformed back to the space of the deformable model before being applied, using equation (Eq. 61), and the solution $\mathbf{u}$ verifying the constraints is then determined by

$$\mathbf{u} = \mathbf{u}_{free} + (\mathbf{C}\mathbf{H}^T)\mathbf{r} \quad (64)$$

Substructure-based contact model

The following approach was developed for the computation of real-time deformations of wire-like structures under contact. More specifically, the method was proposed for the simulation of devices such as catheters and guidewires during navigation inside complex anatomical vascular networks. To control the motion of a catheter or guidewire within the vascular network, the physician can only push, pull or twist the proximal end of the device. Since such devices are constrained inside the patient's vasculature, it is the combination of input forces and contact forces that allow them to be moved toward a target. The main characteristics of these wire-like structures is that modeling techniques must enable geometric non-linearities, high tensile strength and low resistance to bending.

As presented in Chapter II, we model such flexible devices (catheter, guidewires) using beam elements, which rely on a non-linear formulation for handling large geometric non-linearities. To solve multiple simultaneous contacts between the flexible device and the vessel wall, direct Lagrange multiplier techniques, penalty forces or quadratic programming approaches will not constrain the flexible device properly. We proposed an approach based on the computation of the Delassus operator $\mathbf{W}$. As mentioned previously, computing $\mathbf{W}$ requires the evaluation of the compliance matrix $\mathbf{C}$. Since $\mathbf{K}$ is computed at every time step to handle the non-linear deformations, the compliance matrix $\mathbf{C}$ is not constant and needs to be efficiently determined. For this we propose an optimization strategy based on substructure decomposition for the computation of $\mathbf{W}$, and a Gauss-Seidel algorithm for handling collision response in situations where the number of contacts points is very large. Contacts are processed from one end of the wire structure to the other one, while accumulating their contribution in the substructure decomposition using the operators $\mathbf{H}$ and $\mathbf{H}^{\mathrm{T}}$. For each node *i* in contact, we compute its local compliance in the contact space defined by $\mathbf{n}_\alpha$, where $\mathbf{n}_\alpha$ is the normal at the contact point. The local compliance of the contact at node *i* is given by

$$\mathbf{W}_{\alpha,\alpha} = \mathbf{n}_\alpha^T(\mathbf{K}^{-1})_{ii}\mathbf{n}_\alpha \quad (65)$$

The computation of $\mathbf{W}_{\alpha,\alpha}$ is very efficient since $(\mathbf{K}^{-1})_{ii}$ is already computed by the substructure analysis. To account for the coupling between contacts, each contact point is processed sequentially from one end of the wire structure to the other one, while accumulating its contribution in the substructure decomposition using the operators $\mathbf{H}$ and $\mathbf{H}^{\mathrm{T}}$. The resulting system is solved using an adaptation of a Gauss-Seidel algorithm (Duriez *et al.*, 2006a). Assuming that M is the total number of

contacts and N the total number of nodes, introducing the substructure analysis in the Gauss-Seidel algorithm reduces the complexity of one iteration from $\mathcal{O}(M^2)$ to $\mathcal{O}(M+N)$ for an identical result.

As a consequence, results for a 100-node model show a computation time of 25 ms for one time step . This includes the computation of $\mathbf{K}$, the substructure analysis, the collision detection, and the computation of the contact response. These results were measured on a Pentium 4 2.6 GHz processor. An example of catheter navigation through a vascular network is illustrated in Figure 35.

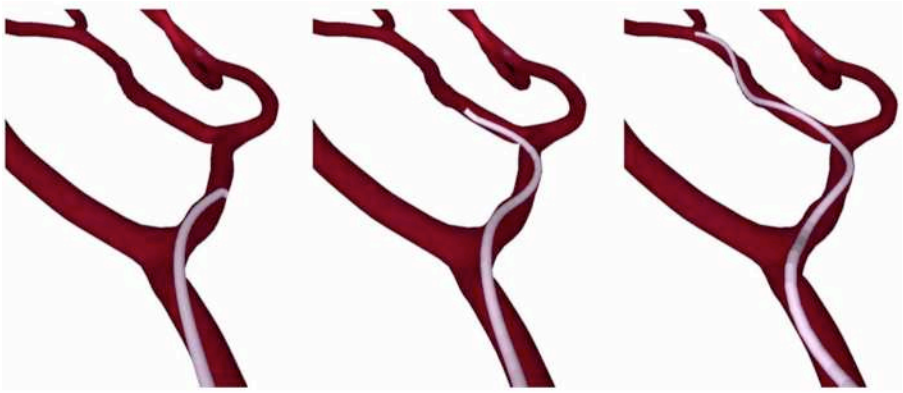


***Figure 35**:* Sequence of images taken from an interactive simulation of catheter navigation. The catheter is constantly in contact with the inner surface of the vessel wall, thus requiring particular care in the choice of the contact method.

The method described above was accepted for oral presentation at MICCAI 2005 (Cotin *et al.*, 2005) and selected for a special issue of Computer Aided Surgery (Duriez *et al.*, 2006a).

Implicit contact model

The Lagrange multipliers technique is a well known mathematical method to define bilateral constraints. Although there exist a few references of work using Lagrange multipliers to solve unilateral constraints, such as (Lenoir *et al.*, 2004) for instance, using Lagrange multipliers for handling complex simultaneous contacts has never been proposed. In the same context as the work described above (simulation of interventional radiology) , we proposed a novel approach and also demonstrated that this approach was equivalent to a more classical mechanics approach (using Signorini's law and Delassus operator).

If we assume that several objects are in contact (these objects can be deformable, rigid or even inert) then we can define a mechanical system representing this set of objects. Whether it is static or dynamic, the stiffness or mass matrix of the system will have a similar structure, i.e. a block diagonal matrix where each block is the stiffness or mass matrix of an object within the mechanical system. Without lack of generality, lets assume the system is static, and that its stiffness matrix is $\mathbf{K}$. In the absence of contacts, each block of $\mathbf{K}$ is independent of the other ones. When contacts are detected, we introduce Lagrange multipliers in the system, thus creating a dependency between certain degrees of freedom. Then, if we use the same decomposition of the motion as previously ($\mathbf{u} = \mathbf{u}_{free} + \Delta\mathbf{u}$), the contact problem can be described as

$$\begin{cases} \mathbf{K}\mathbf{u}_{free} = \mathbf{f} \\ \mathbf{K}\Delta\mathbf{u} = \mathbf{H}^T\lambda \\ \mathbf{H}(\mathbf{u}_{free} + \Delta\mathbf{u}) = \delta \end{cases} \Leftrightarrow \begin{cases} \mathbf{K}\mathbf{u}_{free} = \mathbf{f} \\ (\mathbf{H}\mathbf{K}^{-1}\mathbf{H}^T)\lambda = \delta - \mathbf{H}\mathbf{u}_{free} \\ \Delta\mathbf{u} = \mathbf{K}^{-1}\mathbf{H}^T\lambda \end{cases} \quad (66)$$

Two steps are required to solve these equations. First we compute the free motion from the explicit forces ($\mathbf{u}_{free} = \mathbf{K}^{-1}\,\mathbf{f}$). Given $\mathbf{u}_{free}$, we perform a collision detection that allows us to evaluate $\mathbf{H}$, and therefore $\boldsymbol{\delta}$. We then solve

$$(\mathbf{H}\mathbf{K}^{-1}\mathbf{H}^T)\lambda = \delta - \mathbf{H}\mathbf{u}_{free} \quad (67)$$

and obtain $\boldsymbol{\lambda}$. Since we are dealing with unilateral constraints, not all constraints are necessarily needed to enforce the inequality condition $\boldsymbol{\lambda} \geq 0$. Redundant constraints (for which the corresponding value in $\boldsymbol{\lambda}$ is negative) are then deactivated. This is the so called *status method*. At this point, we can evaluate the corrective motion $\Delta\mathbf{u} = \mathbf{K}^{-1}\,\mathbf{H}^{\mathrm{T}}\,\boldsymbol{\lambda}$ and compute the new position $\mathbf{u} = \mathbf{u}_{free} + \Delta\mathbf{u}$. However, this new configuration does not necessarily meet the initial constraints since we use a linear approximation of the local shape at the point of contact. As a consequence, an iterative scheme is introduced, during which a collision detection is performed on the new configuration to check if an interpenetration exists. If it is the case, a new evaluation of $\mathbf{H}$ is performed, and a new value of $\Delta\mathbf{u}$ is computed. This is repeated until all current contacts are solved, and in most cases, less than 10 iterations and required. Since $\mathbf{K}^{-1}$ does not need to be recomputed during the iterative process, if the collision detection is handled efficiently, these iterations have a reduced computational overhead. At this point, all contacts initially detected are solved. However, when solving these contacts, it is likely that new ones will appear. This is typical of any collision response algorithm. In our case we solve all contacts within a given time step, rather than the next time step. Checking for new contacts within the same time step adds a computational overhead but ensures a more consistent (and contact free) configuration at the beginning of the following time step.

One can note that $\mathbf{H}\mathbf{K}^{-1}\mathbf{H}^{\mathrm{T}}$ describes the coupling between the different contact points, which is exactly the meaning of the Delassus operator defined previously. Moreover, the Lagrange multipliers $\boldsymbol{\lambda}$ give the force in the contact space, which is equivalent to $\mathbf{r}$ in the Delassus operator approach. This means the corrective motion computed using Lagrange multipliers is identical to the one derived from the Delassus operator, i.e. $\Delta\mathbf{u} = \mathbf{K}^{-1}\,\mathbf{H}^{\mathrm{T}}\,\boldsymbol{\lambda} = \mathbf{C}\mathbf{H}^{\mathrm{T}}\mathbf{r}$. The equivalence between Signorini's law / Delassus operator and our approach based on Lagrange multipliers / status method is very important the validity of our approach.

We performed a series of simulations on a Dual Core processor machine with 2 GB of memory and obtained real-time computation rates (25 Hz). These timings include the computation and inversion of the system stiffness matrix K at each time step, as well as collision detection and collision response. Since the contacts are solved in the contact space, the size of the system is the number c of contact (defining n as the number of DOFs, $c \leq n$ and usually $c << n$). It is also important to mention that, in order to enforce the convergence and stability of the contact algorithm, we use a subdivision strategy where each time step is subdivided into a variable number of sub-steps. The initial time step is subdivided if not all contacts have been solved after N iterations of the main loop of the algorithm. This subdivision strategy allows us to solve complex contact configurations and to handle concave cases has a succession of convex cases (see Figure 36).

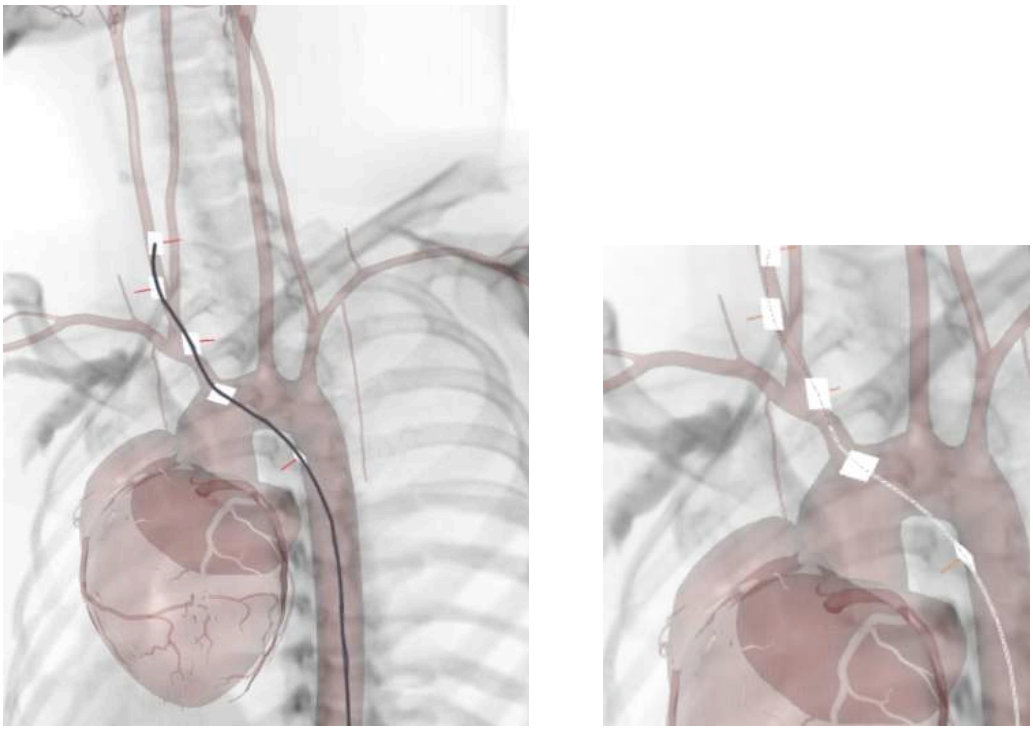


***Figure 36**:* Illustration of interactive simulation of catheter navigation. The catheter is constantly in contact with the inner surface of the vessel wall, which creates complex contact configurations, with simultaneous contacts appearing within the same time step. Our iterative approach guarantees a contact-free configuration at the end of the time step.

The method described above was published in the proceedings of MICCAI 2007 (Dequidt *et al.*, 2007) and is being incorporated within the SOFA framework described in Chapter VI.

Contact compliance warping

Our final and most recent work on contact modeling consists of a generic and very efficient approach for precise computation of contact response between various types of objects commonly used in interactive simulation (medical simulation or computer animation). Our method offers several advantages over previous work. In particular we propose a formalism where an approximated contact model can be derived from the behavior model of the object while verifying Signorini's law of contact and Coulomb's law of friction. This is illustrated on several examples, including deformable models where an approximated compliance matrix is used to estimate the objects' motion required to solve the contact.

The generic approach to contact modeling introduced previously (based on the Delassus operator) can provide stable results even with highly-constrained models. Another advantage of the method is that the non-linear free motion can be computed separately with an adapted solver for each model, as it has been illustrated in the case of flexible medical devices. This part, which can be costly for some deformable objects, can also be parallelized or ported onto the GPU. However, this approach also requires the computation of the compliance matrix $\mathbf{C}$ for each of the colliding objects (or at least, the part of the compliance matrix which involves the colliding nodes). With deformation models discretized with a large

number of nodes, or when many contacts occur, this computation could be very slow. To extend our approach to a large number of deformable models (not only linear models or models which structure allows efficient optimizations) we introduce the idea of approximating the compliance matrix.

The decomposition of the object's motion in a free motion and a corrective motion allows the use of two different behavior models: we can use an approximation of the compliance matrix during the corrective motion while maintaining the accurate dynamic model during the free motion. Then, if the approximation is sufficiently close to the exact compliance matrix, we can use it to compute the Delassus operator $\mathbf{W}$ and the corrective motion $\Delta\mathbf{u}$. This way, even if the motion will be slightly altered, we can still ensure that the corrective motion follows Signorini and Coulomb laws.

A practical example of this approach can be applied to the case of deformations with large rotations. The approximation on the compliance we introduce is only used in the corrective motion step. We propose to compute an approximation of the object's compliance $\mathbf{C}^*$ from the compliance $\mathbf{C}^0$ of the object in its rest position. The approximation is given by

$$\mathbf{C}^*(\mathbf{u}) = \mathbf{R}(\mathbf{u})\mathbf{C}^0\mathbf{R}^T(\mathbf{u}) \qquad (68)$$

Where $\mathbf{R}(\mathbf{u})$ is a 3x3 block diagonal matrix that gathers the rotation associated with object nodes. This simplification speeds up the computation of the compliance needed in the time stepping scheme, because the matrix $\mathbf{C}^0$ could be pre-computed. Such an approximation shares an analogy with deformable co-rotational models (Felippa, 2000). To compute a corrective motion coherent with the LCP, we compute the corrective motion using the same compliance matrix. The exact steps involved in the corrective motion are the following

- Contact forces are mapped to the original coordinate frame of the object: $\mathbf{f}_c^0 = \mathbf{R}^T\mathbf{r}$
- The corrective motion is computed in the original coordinate frame: $\Delta\mathbf{u}^0 = \mathbf{C}^0\mathbf{f}_c^0$
- The corrective motion is transformed (rotated) to the current coordinate frame: $\Delta\mathbf{u} = \mathbf{R}\Delta\mathbf{u}^0$

This method is efficient since only a small number of nodes are usually involved in the contact, which results in $\mathbf{f}_c^0$ being highly sparse. Moreover, the displacement computation provides a perfect correction of the detected interpenetration and follows Signorini's. Nevertheless, as a consequence, the corrective motion is not completely based on the object constitutive law. However, small error on deformation are visually less disturbing that error on interpenetration. Moreover, the approximation is partly corrected by the free motion, based on exact constitutive law, of the next time-step.

We performed a series of simulation test on an Intel Celeron M 520 at 1.6 GHz with 1Gb of RAM using various objects in several scenarios involving different contact configurations. Performances were measures and are reported in the table below (Table 3).

Number of tetrahedra	Number of contacts	Exact Compliance Computation time	Approx. Compliance Computation time
862	35	13,794 ms	8.5 ms
862	13	6,695 ms	1.5 ms
862	36	16,908 ms	9.3 ms
862	12	8,203 ms	1.3 ms
1724	67	11,030 ms	4.7 ms

***Table 3**:* comparison chart of computation times using an exact method for computing the compliance matrix, and our method based on an approximation. Our approach is more than 1,000 times faster.

The object deformation is based on elasticity theory and uses a co-rotational approach (Felippa, 2000). The table below illustrates our results, and clearly shows that computation times obtained using our method do not depend on the complexity of the model used to simulate the object deformation. If the same number of contact points are involved, our method is as fast on a simple model than on a complex one. On the opposite, using the exact requires to solve a system that depends on the model complexity, thus leading to much longer computation times (Table 3).

The method described above was accepted for oral presentation at the 2008 Computer Graphics International conference (Saupin *et al.*, 2008a).

4. Haptic feedback

Haptic feedback certainly plays a role in medical simulation, although it is not quite clear how important this role is. It obviously depends on the type of procedures, and for instance it would be difficult to conceive a simulation of orthopedic surgery without force feedback. In other procedures, such as micro-surgery or interventional radiology, the physician experiences nearly no force feedback (it is more likely to be tactile feedback). In (Batteau *et al.*, 2004) the authors report a study regarding human sensitivity to haptic feedback, in particular the ability of individuals to consistently recall specific haptic experience, and their ability to perceive latency in haptic feedback. Results suggest that individual performance varies widely, and that this ability is not correlated with clinical experience. A surprising result was the apparent insensitivity of test subjects to significant latency in haptic feedback. Overall, it appears that in a very large number of procedures, vision remains the most important feedback.

Providing haptic feedback in a simulation requires at least three elements: a haptic interface, a more or less complex control loop, and a computation of the forces to be sent to the device. In this section, a brief presentation of haptic interfaces relevant to medical simulation is given, followed by a presentation of different approaches for computing interaction forces, known as haptic rendering.

Haptic devices

Haptic feedback can be broadly divided into two modalities: vibrotactile and kinesthetic. Vibrotactile feedback stimulates human subcutaneous tissue. It has been employed in mobile phones, video console gamepads, and certain touch panels. Kinesthetic feedback focuses on the gross movement of the human body. It has been employed in medical simulation trainers, programmable haptic knobs, video game steering wheels, and virtual reality systems. “Force feedback” is most often used to describe kinesthetic feedback.

Among the commercially available haptic devices, the most frequently reported in the field of medical simulation are the PHANToM (SensAble Technologies Inc.), the Laparoscopic Impulse Engine (Immersion Medical, Gaithersburg, Maryland, USA), the IHP, CHP and VSP (Mentice Corporation / Xitact S.A.). While the PHANToM devices are generic solutions, the Laparoscopic Impulse Engine and IHP are designed for laparoscopic simulation, and the CHP or VSP are dedicated to peripheral interventions and interventional radiology (see Figure 37). These different devices have been integrated in several prototypes, and a few commercial products.

<table>
<tr><td>  </td><td>  </td><td>  </td></tr>
<tr><td>PHANToM Premium</td><td>PHANToM Omni</td><td>Xitact CHP</td></tr>
<tr><td>  </td><td>  </td><td>  </td></tr>
<tr><td>Xitact IHP</td><td>Laparoscopic Surgical Workstation</td><td>Laparoscopic Impulse Engine</td></tr>
</table>

***Figure 37**:* Some of the commercially available haptic devices frequently reported in the literature on medical simulation. Some of these systems are integrated in commercial simulators.

Among the new technologies (commercially available or not) that could be relevant for medical simulation, we can mention a haptic interface from Mimic Technologies (Seattle, USA) which differs significantly in its design from the devices mentioned previously (see Figure 38). Also, the Cubic, a haptic interface based on a parallel mechanism, other a slightly larger workspace, more accurate position tracking and higher force restitution. Finally, the Freedom 7S is a haptic interface based on a serial mechanism, developed at McGill University, and equipped with grippers, with applications in surgery simulation (Figure 38).

Mantris from Mimic

Cubic from MPB Technologies

Freedom 7S - McGill University

***Figure 38**:* Examples of new designs of haptic interfaces which could be used in medical simulation applications.

An area of medicine that remains very challenging for medical simulation when it comes to haptic feedback is open surgery. The larger workspace, the absence of ports to guide the instruments, make the design of such interfaces very difficult. A recent report (Hu et al., 2006) presents the development of an interesting untethered haptic feedback system for open surgery simulation. Developed by Energid Technologies (Cambridge, MA, USA) the system uses novel vision-based tracking system and a new haptic device capable of applying feedback through magnetic force. The system is still being developed but would eventually allow surgeons to freely use unconnected knives, clamps, forceps, needle holders, and scissors during a simulation.

In our work we have used the Laparoscopic Impulse Engine for the development of laparoscopic training system, the VSP for the work on interventional radiology, and we developed a custom tracking and haptic device during our early work on interventional cardiology (see Chapter V). While designing a haptic device is very challenging, interfacing commercially available systems to a simulator is usually straightforward. On the other hand, computing forces resulting from tissue-tool interactions at a very high frequencies, without creating instabilities in the system, is far more difficult.

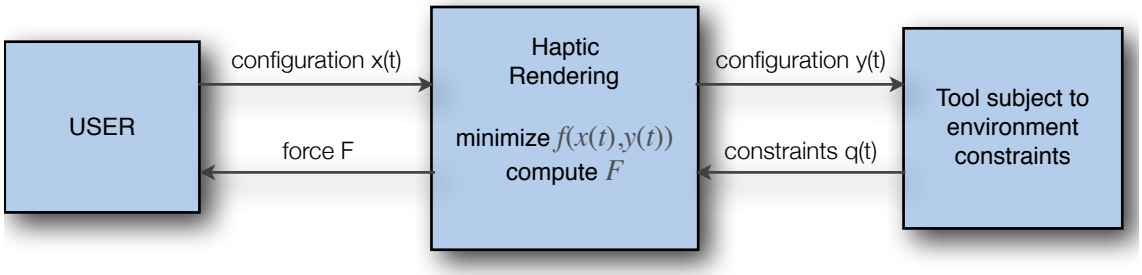
Haptic rendering

Haptic rendering is given by a combination of algorithms which get the position value from the device and compute the torques that needs to be applied by the motors on the haptic device. A part of these algorithms are completely related to classical robotics, i.e. the geometrical direct and inverse model of the device will give the position-force relations from the motors to the end-effecter that is driven by the user. We will not develop this here, and will essentially focus on the following two aspects: 1) the computation of the haptic rendering through the interaction of the manipulated tool with its environment, and 2) the control law of the robotic system which ensures transparency and stability. As we have seen in this

chapter, the main components required to model an interaction (and therefore to produce a haptic rendering) are: a collision detection module, a model of the instrument being manipulated, and a solver that determines the haptic force $F(t)$ and the corresponding configuration of the tool. This solver relies essentially on the choice of an appropriate contact model.

Definition

We consider the rendering problem as defined in (Lin *et al.*, 2008): given a configuration of a haptic device $x(t)$, find a configuration of the tool $y(t)$ that minimizes an objective function $f(x(t)-y(t))$, subject to environment constraints. Display to the user a force $F(x(t), y(t))$ dependent on the configurations of the device and the tool. This definition assumes, somehow, that the input variable is the configuration of the haptic device x, and the output variable is the force F. Contacts between the virtual tool (driven by the user through the haptic interface) and the environment will constrain the tool's position. Then, the haptic rendering will compute forces depending on these constraints to produce a suitable configuration for the tool. This causality between configuration of the haptic device x, and the output variable is the force F is known as impedance rendering, because the haptic rendering algorithm can be regarded as a programmable mechanical impedance (Adams *et al.*, 1998). In impedance rendering, the device measures the positions of the end effecter (which is driven by the user) and the algorithm computes a force that is rendered to the user by the device (see diagram below). A different possibility is admittance rendering, where the haptic rendering system can be regarded as a programmable mechanical admittance that computes the desired device configuration, as the result of input device forces. But pure admittance rendering necessitates a force captor on the device which is not usually available.



In the definition of the haptic rendering problem, the objective function that must be optimized for computing the tool's configuration and the function for computing the forces transmitted to the user represent the main algorithms developed for haptic rendering.

Tool Model

In medical simulation, the choice of the tool model depends on the application. For instance, for a laparoscopic instrument, a rigid articulated model with few degrees of freedom (DOFs) is adapted, but for interventional radiology, where the device is a catheter navigating inside the vessels, the tool model is obviously more complicated. As the model is incorporated in the haptic rendering algorithm, it has to be computed at very high rates. Several optimization strategies propose to use a simpler (bio)mechanical model for the haptic rendering than the one used in the simulation.

Collision Detection

In the context of haptic rendering, collision detection is the process that, given a configuration of the virtual tool, detects potentially violated environment constraints. Collision detection can easily become the computational bottleneck of a haptic rendering system with geometrically complex objects, and its cost often depends on the configuration space of the contacts

Collision Response

In algorithms where the tool's configuration is computed through a dynamic simulation, collision response takes the environment constraints $q(t)$ given by the collision detection module as input and computes forces acting on the tool. Collision response is tightly related to the formulation of environment constraints $q(t)$ and can be implemented in many ways, as described in previous sections.

Haptic coupling

Haptic rendering algorithms provide constrained configurations for the instrument being manipulated and associated constraint forces. These forces have to be rendered by the haptic device, i.e. a robotic arm. As soon as some forces are applied through the interface, the user, haptic interface, and simulation create a complex mechanical system that has to be controlled to avoid instabilities.

Direct Rendering Algorithm

Direct rendering relies on an impedance-type control strategy. First, the configuration of the haptic device is received from the controller, and is assigned directly to the virtual tool. Collision detection is then performed between the virtual tool and the environment. Collision response is typically computed as a function of object separation or penetration depth using penalty-based methods. Finally, the resulting contact force (and possibly torque) is directly fed back to the device controller.

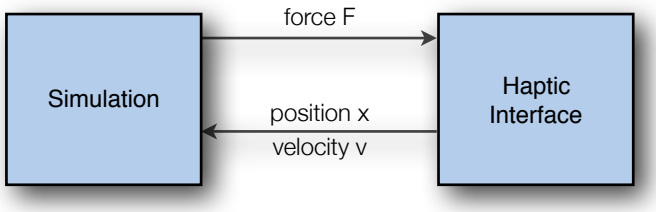


***Figure 39**: Direct rendering principle: the simulation integrates the controller for the haptic device. Since the two are not dissociated, this requires the simulation to run at very high frequencies (at least hundreds of Hertz).*

The popularity of direct rendering stems obviously from the simplicity of the calculation of the tool's configuration, as there is no need to formulate a complex optimization problem (e.g. rigid body dynamics model for 6-DOF haptic rendering). However, the use of simpler approaches, such as penalty methods for force computation has its drawbacks, as penetration values may be quite large and visually perceptible, and system instability can arise if the force update rate drops below the range of stable values. Moreover, the simulation and the device controller are not dissociated, thus requiring the simulation to act as a controller for the haptic device. If haptic rates of 300 Hz or more are required, as it is often the case, this creates very high computational constraints on the simulation as it also needs to run at 300Hz or even higher frequencies.

Rendering through virtual coupling

Despite the apparent simplicity of direct rendering, the computation of contact and display forces may become a complex task from the stability point-of-view. Stability enforcement can largely be simplified by separating the device and tool configurations, and inserting in-between a viscoelastic link referred to as virtual coupling (Colgate *et al.*, 1995). The connection of passive subsystems through virtual coupling leads to an overall stable system. Contact forces and torques are transmitted to the user as a function of the translational and rotational misalignment between virtual tool and device configurations. The most common form of virtual coupling is a viscoelastic spring damper link. Such a virtual coupling was used by (Zilles et al., 1995) (Ruspini *et al.*, 1997) in the "god-object" and "virtual proxy" algorithms for 3-DOF haptic rendering. The concept was later extended to 6-DOF rendering (McNeely *et al.*, 1999), by considering translational and rotational springs. The use of a virtual coupling allows a separate design of the impedance displayed to the user (subject to stability criteria) and the impedance (i.e. stiffness) of environment constraints acting on the tool. Environment constraints can be of high stiffness, which reduces (or even completely eliminates) visible interpenetration problems. As we cannot anticipate the position of the haptic interface, the role of the simulation is to provide a corrected position and velocity of the virtual tool in the simulation. If the simulated tool is constrained by its environment, position and velocity values compatible with these constraints will be computed. However, if the object is completely free, the position and velocity of the device will be directly applied to the virtual instrument in the simulation. We obtain the following scheme:

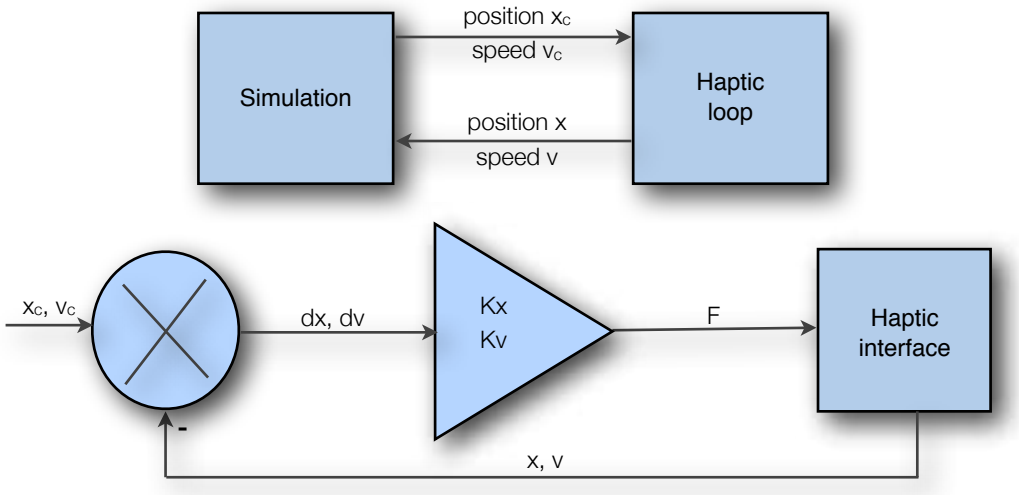


***Figure 40**: general structure of a rendering algorithm based on virtual coupling (for impedance rendering). The input to the rendering algorithm is the device configuration, but the tool configuration is solved in general through an optimization problem, which also accounts for environment constraints. The difference between device and tool configuration is used both for the optimization problem and for computing output device forces. Xc et Vc are, respectively, the corrected position and corrected velocity. The corresponding control law for the haptic interface is based on a proportional derivative (PD) controller.*

Virtual coupling algorithms may suffer from undesirable filtering effects, in case the update rate of the haptic rendering algorithm becomes too low, which highly limits the value of the rendering impedance. Multi-rate algorithms (Otaduy *et al.*, 2006) can improve the transparency of the rendering by allowing stiffer impedances. We used such an approach in (Saupin et al., 2008b) to simulate haptic interactions between a laparoscopic grasper and the liver. To reach this goal, we proposed a novel, generic and very

efficient approach for precise computation of the interaction between organs and instruments. The method includes an estimation of the contact compliance of the contact zones of the organ and of the instrument. This compliance is then used as an approximated model (see Figure 41) by a multithreaded local haptic model. Contact computation is performed in both simulation and haptic loops, according to advanced contact models. Realistic and stable interactions on non-linear models are possible using an implicit time integration scheme (see Figure 42).

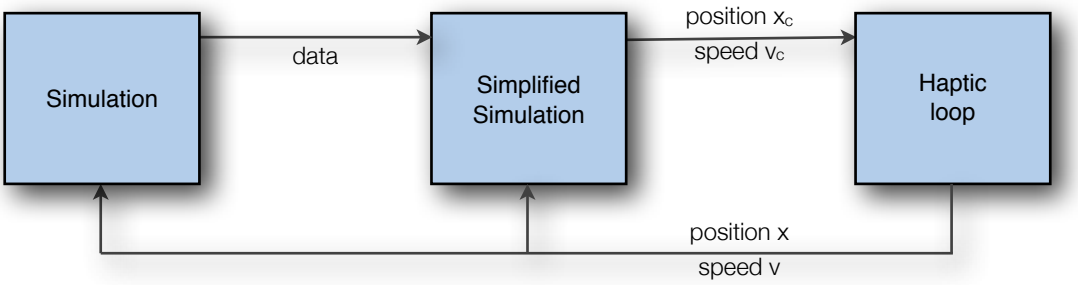


***Figure 41**: Local haptic models are often used to compute haptic feedback at a higher frequency than the main simulation loop. The local or simplified model needs to provide information close to the actual simulation.*

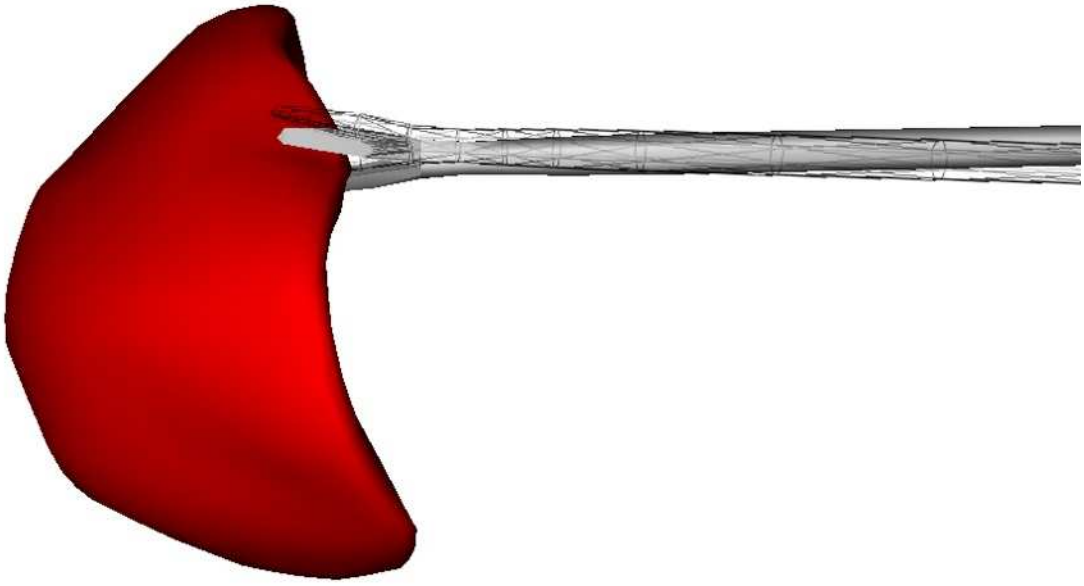


***Figure 42**: An illustration of our work on haptic rendering for laparoscopic surgery. A simplified, very efficient model is used to compute the forces while the deformation of the liver uses the "full" deformable model. The liver model is a co-rotational FEM model allowing for large displacements, while the haptic model uses a simplified version (linear) to compute forces at a very high frequency. The result is a visually correct interaction between the laparoscopic instrument and the organ, without any interpenetration.*

Summary of contributions

- **Research articles**
  - J. Rabinov, S. Cotin, J. Allard, J. Dequidt, J. Lenoir, V. Luboz , P. Neumann, X. Wu, and S. Dawson. “EVE: Computer Based Endovascular Training System for Neuroradiolgy”. ASNR 45th Annual Meeting & NER Foundation Symposium, pp. 147-150, 2007.
  - C. Duriez, S. Cotin and J. Lenoir. “New Approaches to Catheter Navigation for Interventional Radiology Simulation”. In: Computer Aided Surgery, vol. 11, p. 300-308, 2006.
  - S. Cotin, C. Duriez, J. Lenoir, P. Neumann, and S. Dawson. “New approaches to catheter navigation for interventional radiology simulation”. Proceedings of the MICCAI Conference, MICCAI 2005, pp. 534-542, 2005.
  - G. Saupin, C. Duriez, L. Grisoni, and S. Cotin. “Efficient Contact Modeling using Compliance Warping”. In Proc. Computer Graphics International, 2008 (to appear).
  - G. Saupin, C. Duriez, and S. Cotin. “Contact model for haptic medical simulations”. In Proc. International Symposium on Computational Models for Biomedical Simulation - ISBMS, 2008 (to appear).
  - J. Dequidt, J. Lenoir, and S. Cotin. “Interactive Contacts Resolution Using Smooth Surface Deformation”. Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), LNCS 4792, pp. 850-857, 2007.
- **Patents**
  - Cotin S, Wu X, Neumann P, Dawson S; "Methods and Apparatus for Simulation of Endovascular and Endoluminal Procedures", United States Patent Application, 60/600,188 - PCT application number PCT/US2005/028594, 2005.
- **Software**
  - **EVE: interventional radiology training system**: different algorithms for collision response and collision detection have been integrated in the EVE system.

Chapter IV — Validation

1. Introduction

An essential aspect of a simulation is its ability to predict a given behavior based on a set of input parameters and constraints. Yet, as we have seen previously, the variety of approaches, along with the simplifications that are required to achieve real-time performances, often lead to models that provide, at best, visually plausible results in a specific context. Quantitatively measuring the accuracy of a model and its range of validity is essential for the development of medical simulation as a tool for the medicine of the 21st century. But validation should not be limited to assessing biomechanical models. It should be applied to any essential element of a simulation system, such as anatomical models, medical devices, and of course the overall training or planning system.

Computational techniques for the analysis of mechanical problems have recently moved from traditional engineering disciplines to biomedical simulations. Thus, the number of complex models describing the mechanical behavior of medical environments have increased these last years. While the development of advanced computational tools has led to interesting modeling algorithms, the relevances of these models are often criticized due to incomplete model verification and validation. An objective of our work is to propose a methodology for assessing deformable models. Computational tools need to be developed for assessing the accuracy and computational efficiency of new modeling algorithms proposed in the context of medical simulation. To this end, a set of metrics need to be defined, as well as reference models such as biomechanical phantoms or analytical models. The same principles also apply to other elements in a simulation, such as algorithms for flexible devices or anatomical models.

However, evaluating the algorithmic capabilities of a simulator is not the only requirement for guaranteeing its fidelity or efficiency as a training system. To assess the overall quality of a training system, other metrics need to be defined, and validated. These metrics should measure various characteristics of the simulator, such as face, construct or concurrent validity. Such validations are already being performed on a few commercially available systems, and are highly expected by the medical community.

Throughout the following sections we will describe our contributions in various areas of validation, in the context of medical simulation. These include: validation phantoms and metrics for anatomical modeling, validation phantoms and metrics for soft-tissue models, and performance metrics for skills training.

2. Validation of anatomical models

Validation models or phantoms have often been used to measure the accuracy of image processing techniques. When segmentation techniques became efficient enough that they would give results close to a manual segmentation, or when manual segmentation would become potentially biased, using phantoms became a more reliable solution. Phantoms have generally been developed to assess the accuracy of segmentation algorithms. Accuracy criteria are usually derived from distance-based

discrepancy measures between the result of the segmentation and the phantom model, taken as a reference. Furthermore, segmentation errors can be spatially determined, and sorted based on their distance to the reference model. Noise can also be added to this initial data to determine the sensitivity of the segmentation algorithm to noise, often present in medical images.

Anatomical phantoms can also be used to assess the quality of the reconstruction process. While three-dimensional reconstruction from segmented data is usually not the main challenge in medical imaging, it is an important aspect of medical simulation, as we have seen in Chapter II. Assuming a segmentation of a volumetric data set has been obtained, there exist different ways of reconstructing a surface model from this data. Phantoms can therefore be useful to quantitatively evaluate the difference between various techniques, or the influence of parameters. Compared to using synthetic data sets, they allow recreating imaging conditions that are closer to actual clinical setups.

Vascular phantom

During the course of our research on interventional radiology simulation, we developed a vascular phantom designed using silicone gel and flexible tubing. Initially designed for the validation of the catheter model, it was eventually used to quantitatively measure the quality of the segmentation and reconstruction method described in Chapter II. This phantom is composed of a Plexiglas box, inside of which nylon tubing represents a simplified vascular network, with vessels of different sizes (Figure 43). Silicon covers the tubing therefore giving them a slight rigidity and a protection. The radii of the vessels are variable, starting with a 2.34 mm radius (to reproduce the middle cerebral artery) and ending with seven vessels of radius ranging from 0.78 mm to 1.17 mm.

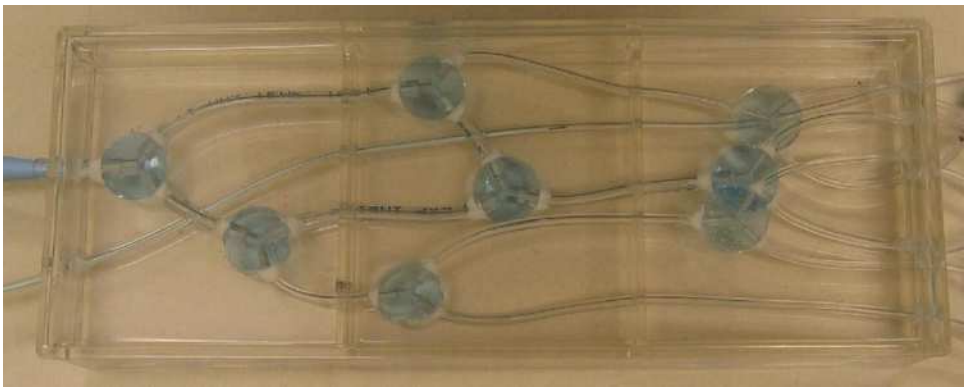


***Figure 43**: custom-designed vascular phantom, made of nylon tubing embedded in silicon gel.*

To evaluate the rotational invariance and robustness of our method, we injected a contrast agent inside the phantom which was later scanned this phantom in 12 different positions. Most of those are obtained via a rotation of 45 degrees of the phantom on one or more axis. The resolution of the data sets is 0.6 x 0.6 x 1.25 mm (for the x, y, and z axis). Following the scans, a segmentation algorithm was applied to each of the 12 data sets, using the skeletonization method described in Chapter II. The lengths and radii of the computed skeletons were analyzed through the Bland-Altman method (Bland et al., 1986). In 100% of the cases, the length variation remained within two standard deviations. In 97% of the cases, the radius variation remained within two standard deviations. This characterizes the excellent accuracy and robustness of our method.

Three-dimensional reconstruction

Surface smoothness was then measured as well as the distance between the surface reconstructed with our method and the surface reconstructed with a standard threshold followed by a marching-cubes algorithm (Lorensen *et al.*, 1987). The result of the marching cubes was used as the reference surface. Smoothness is determined as the Root Mean Square of the minimal and the maximal surface curvatures. Distance between the isosurface and the result of our reconstruction algorithm were computed and compared to the distance of the isosurface after applying a decimation algorithm (*vtkDecimatePro* from the VTK[4] library). The Root Mean Square error is always less than one voxel (0.6 mm) and lower than 0.4 mm after one level of subdivision. Moreover, the smoothness of our reconstructed surface is almost constant for all subdivision levels and at least 10 times lower than any Marching Cubes surface, whether it was followed by a decimation step or not. Consequently, our process allows simpler, more regular meshes to be generated, with reasonable error and good smoothness (Figure 44).

This work on the validation of the three-dimensional reconstruction process was preliminary, and limited to the case of vascular structures. Yet, it illustrates the influence of the choice of the reconstruction technique in the overall result, and that this result is not only a consequence of the segmentation process. More importantly, this work also illustrates that optimized models, compatible with the constraints of interactive simulation, can be as accurate as conventional methods. This is an important element, as we move towards patient-specific simulations.

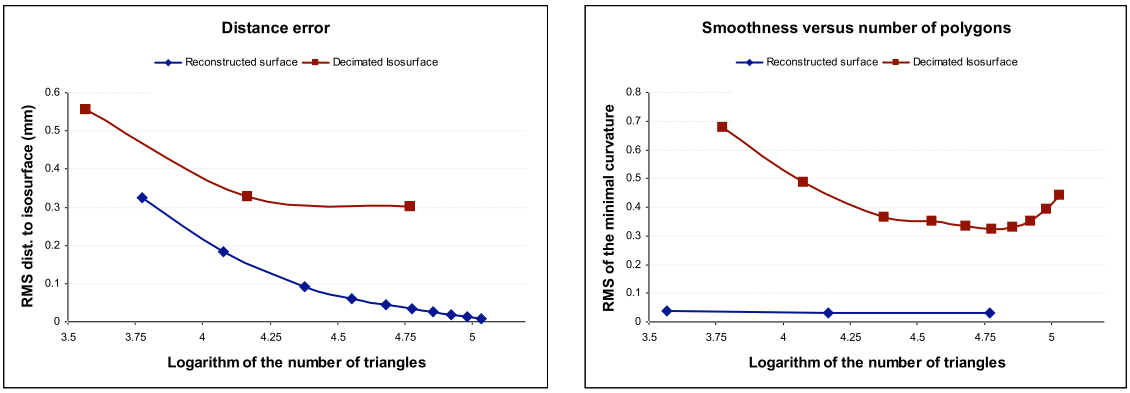


***Figure 44**: (Left) Comparison of the root mean square distance error on a phantom data set vs. the number of triangles. The original isosurface was decimated using the vtkDecimatePro algorithm. (Right) Evolution of the smoothness by applying vtkSmoothPolyDataFilter from 0 to 40 iterations compared to the smoothness of our reconstructed model.*

3. Validation of soft tissue models

As we have seen previously, a key objective in medical simulation is to create accurate biomechanical models of anatomical structures that exhibit the main characteristics of the soft-tissues under deformation. The resulting constitutive laws are often non-linear, viscoelastic, possibly anisotropic, or

[4] VTK: Visualization Tool Kit (http://public.kitware.com/VTK/)

involving non-Newtonian porous flow, and other complex phenomena. We have presented an iterative scheme for determining model parameters to minimize an error function between the model and experimental data. Although this permits to validate (to a certain extent) the model, this validation remains traditionally limited to force-displacement data. Three-dimensional assessment of the predictive capabilities of the model is an area that has not been addressed so far, yet it is essential to validate biomechanical models in the context of interactive medical simulation. In addition, when real-time algorithms are developed, they often rely on assumptions and simplifications of the actual constitutive law, and in some instances, they are not even based on biomechanical principles at all.

The overall objective of a validation process is to guarantee that: (i) the numerical approximation of the mathematical equations chosen for governing the model is acceptable and (ii) the model provides an accurate representation of the physical behavior of the problem of interest within a given computation time. Both assumptions need to be verified within an assessment of error in the model predictions and their achievement relies on a combination of methodologies and experimental data. A review on verification, validation and sensitivity studies has recently been proposed in the context of computational biomechanics (Anderson *et al.*, 2007). In their paper, the authors present the concepts of verification and validation of biomechanical models and introduce a guide to perform such studies. In the context of medical simulation, only a few authors have proposed ideas for testing, comparing and quantifying the results of modeling methods, in particular soft tissue models. Alterovitz *et al.* (Alterovitz *et al.*, 2002) have suggested accuracy metrics and benchmarks for comparing different algorithms based on the Finite Element Method. Validations procedures for discrete approaches have also been introduced (Bianchi *et al.*, 2004) (Baudet *et al.*, 2007). However these studies mainly focused on the identification of parameter sets that optimize the accuracy of the discrete models. Real data can also be used as reference models and experimental results have already been presented in the context of medical simulation. Among them, the Truth Cube experiment (Kerdok *et al.*, 2003) or experiments on cylinders (Leskowsky *et al.*, 2006) offer quantitative results, allowing the comparisons of modeling methods with real three-dimensional data. A comparison of FEM simulations with medical images has also been proposed in (Chabanas *et al.*, 2004).

While these different works propose interesting ideas for defining reference models or for establishing metrics to compare and assess soft tissue models, no consensus has been found on a common methodology. Similarly, data for assessing soft tissue models is rarely publicly available. As more and more models are being proposed to achieve real-time deformations (see Chapter II), it seems critical to establish a methodology (i.e. a set of protocols, metrics, and reference models) to better determine the range of validity and application spectrum of currently available algorithms. To support the model validation effort, and provide a set of gold-standard data (also referred to as biomechanical models) to the wider deformable-object modeling community, we created the Truth Cube and a series of other standard test objects. Work on the Truth Cube was published in the journal MedIA (Kerdok *et al.*, 2003). Work on cylindrical phantoms was published in (Valtorta *et al.*, 2005). Very recently we also started a reflexion on a methodology for assessing and comparing algorithms for (real-time) soft tissue models. The driving idea behind this work is to propose new biomechanical phantoms with material characteristics closer to actual soft tissues, but mostly to propose an open framework for validating the algorithms (Marchal *et al.*, 2008).

Biomechanical phantoms

To assess on one hand the predictive nature of complex constitutive models, and on the other hand the accuracy of real-time algorithms for deformable structures, we need to rely on simple, well characterized materials and shapes. Also, we believe it is important to validate models against real world objects rather than against other models, such as advanced finite element models. While there is no significant difference between a finite element model and the real object in the case of a simple material (linear elastic for instance), the difference might be much larger when the constitutive model increases in complexity.

Truth Cube

The original Truth Cube was a clear silicone cube of size 8x8x8 cm^3, constructed in layers with Teflon beads embedded in a regular grid with 1 cm spacing (see Figure 45). The silicone gel used for most of the Truth Cube was characterized using a static material testing machine. The material properties of the cube were found to be isotropic and linear with a Young's modulus of 15 kPa and an assumed Poisson's ratio of 0.499. Although a phantom based on a linear material is not our final objective (most soft tissues have non-linear constitutive laws) it was simpler to characterize and validate for this initial attempt. A second material test was performed on a sample of the same silicone gel, without the embedded beads. Nearly identical parameters were measured, confirming that the low density of Teflon beads in the cube did not modify the intrinsic properties of the silicone gel.

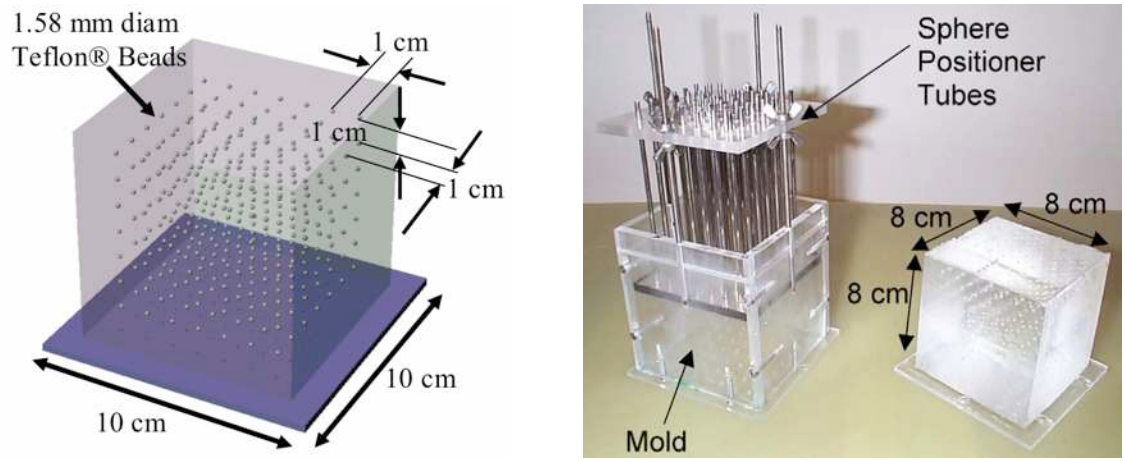


Figure 45: *(Left) Conceptual model of the Truth Cube (right) mold for Truth Cube fabrication, internal sphere positioner, and resulting Truth Cube with embedded markers.*

A series of CT scans with different loading conditions was conducted on the Truth Cube to provide reference data sets for validating deformable algorithms. A first scan was performed on the cube subject only to gravity forces. It was then imaged under uniaxial compression to allow for technique development and result comparison with a fully defined FE model. The uniaxial compression test was accomplished with a thick acrylic plate attached to a vertical support by a low-friction pivot (see Figure 46).

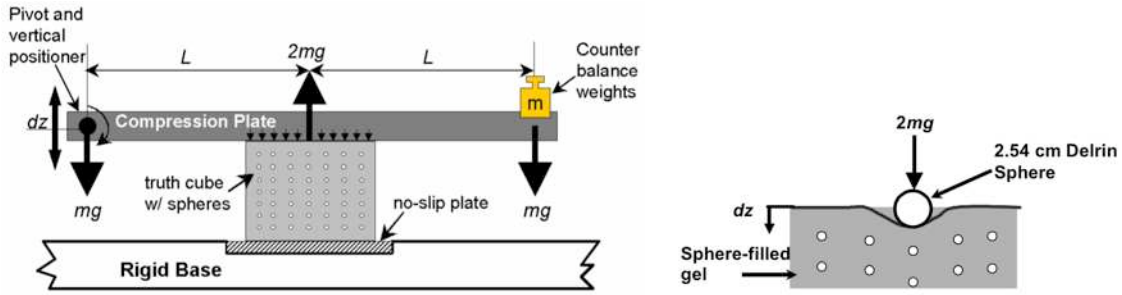


Figure 46: *Experimental setup: (Left) side view of the uniaxial compression setup and close-up schematic of spherical indentation situation. (Top) close-up schematic of spherical indentation situation (Bottom). L is the length from the loading points to the center of the cube, dz is the vertical motion, and m is the mass of the counter weights.*

The plate was loaded by weights applied to the end opposite the pivot (in addition to the weight of the plate itself). The design confined all metal parts (the vertical support rods and weights) to the ends of the apparatus so that they did not interfere with CT scanning of the cube in the middle. A linear dial indicator measured the distance the plate was translated. A similar setup was used for the large deformation spherical indentation test except that a spherical indenter mounted on a cylinder was added to the compression plate (Figure 46). Such a test, using a spherical indenter, was performed to represent the type of deformation created during surgical manipulations.

As a difficult challenge for finite element and other modeling techniques is accurately calculating large deformations, we applied up to 18% uniaxial strain, and 30% nominal hemispherical indentation to the Cube. An analysis of the CT scan data was performed to extract the surface of the cube but also the location of the centroids of the beads embedded within the cube (see Figure 47). The motion of the beads is an important information to determine the internal strain field during the different indentation tests. The resulting data sets were posted on a public website (http://biorobotics.harvard.edu/truthcube), and compared with finite element simulations, using known material properties of the silicone.

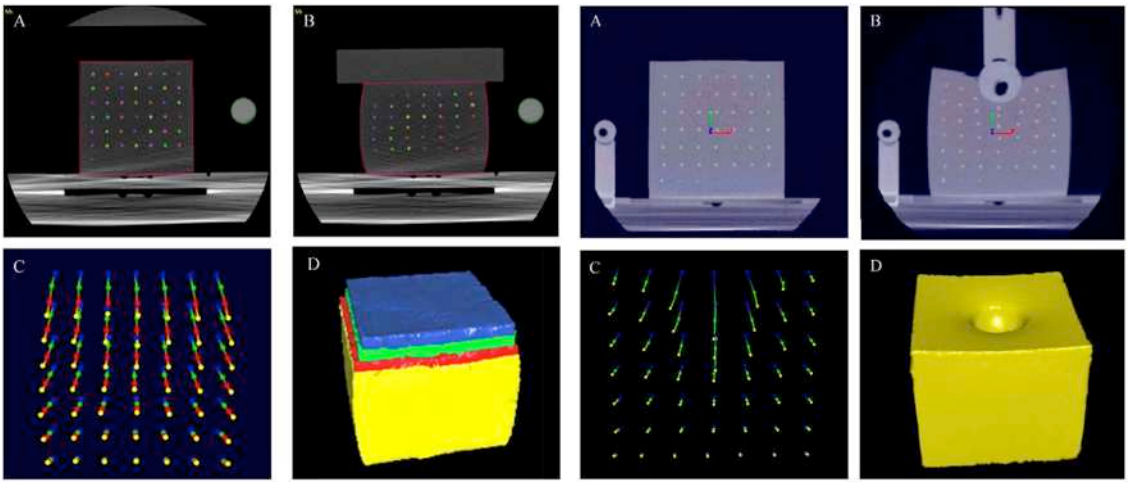


Figure 47: *(Left): CT image of the central vertical plane of the Truth Cube after segmentation in both the unloaded (A) and maximally uniaxial compressed (B) states. Note the registration marker off to the side. Internal sphere trajectory and location in the central plane (C) (blue=unloaded, green=55%, red=12.5%, yellow=18.25% strain). Outer surface of the cube in its four strain states(D). (Right) CT of center vertical slice for spherical indentation in unloaded state (A) and under 30% nominal strain (B). Trajectory and locations of the internal spheres for the same slice is shown in (C) (blue=no indentation, green=22% nominal strain case, yellow=30% nominal strain case). Surface for the 30% strain case is represented in (D).*

For the uniaxial case, large deformation (i.e. nonlinear) analysis was employed using the boundary conditions (frictionless upper surface, fixed lower surface, and free side surfaces) and geometry extracted from the CT scans for the unloaded and 18.25% nominal strain cases. For the spherical indentation case, large deformation (i.e. nonlinear), axisymmetric analysis was employed using the boundary conditions (fixed contact between indenter and upper surface, free side surfaces, and fixed lower surface) and geometry extracted from the CT scans for the unloaded and 22% nominal strain cases. Results from the finite element analysis and experimental data were compared (see Figures 48 and 49) and different error metrics were defined to illustrate the quality of the model.

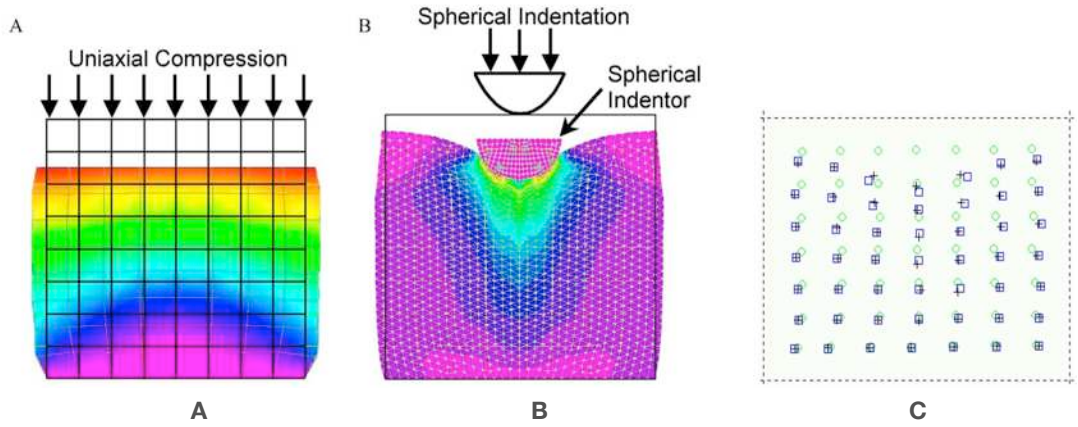


***Figure 48**: **(A)** Center slice of the uniaxial compression coarse mesh under 18.25% strain. **(B)** Center slice of the refined spherical indentation FEM mesh displaying the results of the 22% strain experiment. The magnitude of the strain is coded as color with red indicating the highest strain and violet indicating the lowest. Note that the undeformed model is outlined in black. **(C)** Comparison of the measured internal sphere displacement (boxes) versus the FE model's predicted sphere displacement (crosses) for the 22% nominal strain spherical indentation case. The circles are the original internal sphere locations and the dashed line is the undeformed outer surface.*

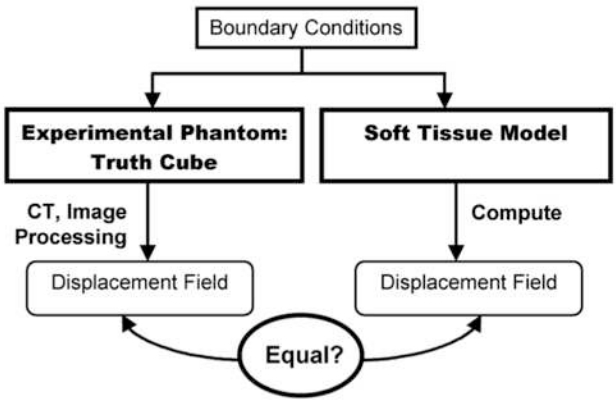


***Figure 49**: Approach for validating soft tissue computational models. Boundary conditions (material property, geometric, loading conditions, etc.) are imposed on the experimental phantom and the model. The displacement fields in the phantom are measured using CT imaging and compared to the model's predicted results.*

This initial study demonstrated the feasibility of experimental measurement of volumetric displacement fields for soft material phantoms under large strains, and provides some useful internal strain field data that can readily be used to validate soft tissue models. The truth cube has regular geometry and well-characterized material properties and loading conditions that was helpful for the development process, but soft tissue simulations must deal with conditions that are vastly different, involving very large deformations, irregular shapes and complex materials. Therefore, other phantoms would have to be designed, such that each of them exhibits some of the key characteristics of soft tissues (non-linearities, visco-elasticity, porosity, etc.).

Truth Cylinder

Subsequent to the testing of the original Truth Cube, collaborators at ETH Zurich (Switzerland), Nagoya University (Japan) and Brown University encouraged the development of a second standard object that could be shared amongst the groups, tested under their own conditions, and compared against all of the others. As the original Truth Cube was destroyed during the final stage of the initial experimentation, we developed a new (somewhat misnamed as its geometry is a cylinder) Truth Cube named TC2 (Figure 50). A detailed analysis of the compiled data was published by Valtorta *et al.* (Valtorta, 2005) providing indications of the scope and applicability of the various testing techniques, and of different methods of simulating the testing methods.

***Figure 50**: (Left) TC2 phantom undergoing torsional resonator testing at ETH Zurich. The phantom is made of Ecoflex 0030 (Smooth-On, Inc., Easton, PA) and includes randomly distributed Teflon beads embedded within it, rather than the regular grid of beads in the original Truth Cube.*

4. Validation of medical devices

Validating soft-tissue models is of course of prime importance in medical simulation. However, as we move towards planning and rehearsal, simulation systems will need to be validated more thoroughly. While some aspects will remain essentially validated through qualitative assessments from clinical specialists, others will require quantitative assessments. This is the case of a large number of medical devices such as stents, coils,, catheters, guidewires, flexible endoscopes, etc.

In the context of a multi-team INRIA project aimed at coil embolization planning, we developed a coil model based on the formulation described in Chapter II. Before integrating the coil model within the training system itself, we designed an experiment to assess the accuracy of the model. The experimental setup consists of a box filled with water and a transversal fixed guide, defining the path for the catheter. The catheter is first introduced, followed by the helical coil (*Micrus MicroCoil Platinum*) as it is done in actual coil embolization procedures. The three-dimensional shape of the coil at different stages of the deployment is obtained from 3D rotational angiographic images using a marching cubes algorithm. The different steps of the experiment are illustrated in Figure 51. In the experiment, the coil is only subject to a gravity force, and contacts with the box were avoided. Detailed characteristics of the coil are difficult to obtained as their are often proprietary information. In this case, the mass, Young modulus, and rest shape geometry are unknown, and need to be measured.

We developed an optimization method, similar to the one illustrated in Figure 49, to determine the unknown parameters. The error criteria is a distance measure between the simulated coil and the 3D reconstructed model (see Figure 51). Using information available from the device manufacturer (length: 150 mm, radius: 0.3556 mm, helix diameter: 7 mm, distance between two turns: 0.25 mm) and using an undeformed helical shape as initial guess, the different parameters of the coil could be identified; mass: 1.28 g, length: 150 mm, radius: 0.3556 mm, helix diameter: 7 mm, Young modulus E = 7.5 GPa, Poisson ratio v = 0.39. These values are consistent with the values given in the handbooks.

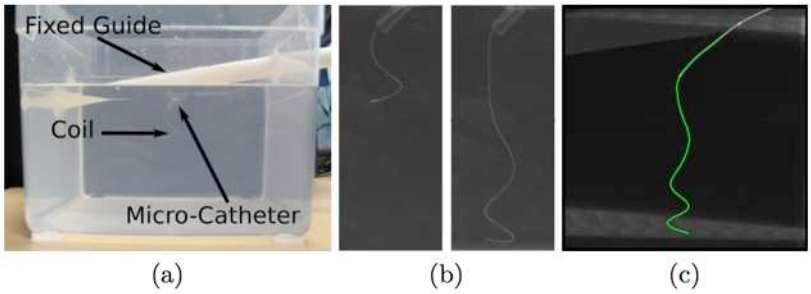
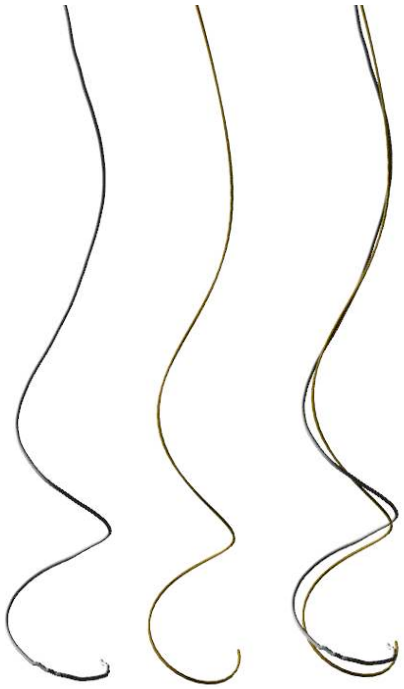


***Figure 51**: Experimental setup: (a) the coil is deployed in a contact-free environment. (b) Volumetric data is obtained by 3DXA and a marching cube is performed to get a mesh of the real coil. (c) The central line of the coil is segmented from the image and a continuous representation using a spline is computed.*

We then used these parameters in a real-time coil model (see Chapter II) and ran a series of experiments to assess the accuracy of the model. Figure 52 shows a visual and quantitative comparison between our simulated coil deployment and a reconstructed model from experimental data. Once the real coil positions have been tracked and the shape segmented for three different stages of the coil deployment, an algorithm using a B-spline formulation was applied to the real coil data to obtain a continuous description of the shape. Then, simulations of the coil were performed using the same boundary conditions as measured during the actual coil deployment. In order to compare our simulation with the actual data, the B-spline curve was discretized using the same number of points as the number of nodes in the simulation. The error metric used to validate the coil simulation is the relative energy norm error (Zienkiewicz, 2000). The metric is defined as the ratio of the displacement between the simulated and the real coil and the real displacement. The error is computed for each node and for three different steps of the deployment. Results are given in Figure 52. The relative energy norm error is very small and shows that our coil simulation, although computed in real-time, gives a configuration that is close to the one of an actual coil.



	% Error (SD) $\frac{\|\mathbf{U}_{real} - \mathbf{U}_{sim}\|}{\|\mathbf{U}_{real}\|}$	Mean displacement $\frac{\|\mathbf{U}_{real}\|}{N_{\text{points}}}$
Stage 1	9.8% (4.05)	18.2 mm
Stage 2	6.19% (5.05)	24.99 mm
Stage 3	4.17% (1.42)	39.11 mm

***Figure 52**: from left to right: reference coil model reconstructed from 3DRA data, simulated coil composed of 100 beam elements, and simulated model overlaid with actual data. The table on the right presents quantitative information about the difference between the simulated model and the real coil. This difference is very small even for large deformations.*

5. Towards a validation methodology

The various ideas, phantoms, and validation tests presented previously were motivated by the same need of assessing the validity of our algorithms, as well as algorithms and models proposed by other researchers. To continued in this direction, a common methodology needs to be established, i.e. protocols, metrics, and a framework. Based on the guide proposed in (Anderson *et al.*, 2007), a protocol for analyzing the performances of a deformable model can be decomposed in two main parts. The first part concerns the verification process of the modeling method. It aims at determining if the model implementation provides a correct description and a solution of the chosen modeling method theory. In this part of the protocol, the benchmarks used to analyze the performances of the model are mainly analytical solutions of well-known problems. Such comparisons have already been proposed in the literature, for example by (Alterovitz *et al.*, 2002). In a second stage called validation, the ability of the already verified model to bring a correct simulation of a real world object has to be guaranteed. In this validation part, computational predictions are compared to experimental data as a gold standard. For both parts of the verification and validation protocol, different types of errors can be identified. The first type concerns numerical errors introduced by solving intractable mathematical equations, among those discretization or convergence errors are very common. This type of error is mainly identified trough the verification process. The second type of error can be called modeling error and is related to assumptions

and approximations in the mathematical representation of the physical problem of interest. Such errors mainly come from geometry representation, boundary condition specifications, material properties or the choice of the governing constitutive equations. They can mainly be measured through the validation process.

While it seems pointless to think of an ideal biomechanical phantom, proposing a validation methodology applicable to large number of models, appears feasible. We have published preliminary work in this direction (Marchal *et al.*, 2008). In this paper we propose to use the SOFA framework (see Chapter VI) as a common framework for 1) integrating newly available algorithms, 2) implementing relevant metrics for assessing the models, and 3) comparing models against phantoms or against other models. To illustrate our proposal, we designed a phantom, implemented a few metrics in SOFA and ran a comparison test across a variety of models. The chosen experiment is an elastic beam under gravity, fixed on one side. This test case is well known in continuum mechanics and has already been used previously for example by (Mueller *et al.*, 2004). For this experiment, an analytical solution is available allowing for a verification procedure. Furthermore, real data experiments have been conducted in order to achieve a validation procedure, thus allowing to perform comparisons against analytical *and* phantom data.

Five different algorithms, available in SOFA, were compared: (a) a linear FEM algorithm with a tetrahedral mesh, (b) a co-rotational FEM algorithm also with a tetrahedral mesh, (c) a co-rotational FEM algorithm with an hexahedral mesh, (d) an algorithm based on beam elements, and (e) a mass-spring network. At the exception of the mass-spring model for which stiffness coefficients were adjusted to obtain the best behavior, the physical parameters used in all simulations correspond exactly to those measured on the experimental data. Concerning the comparisons with the analytical solution, the relative energy norm error can be used as the different meshes of the models and the analytical solution have the same number of nodes. As for comparisons with the experimental data, since no physical markers were used to track volumetric displacements, a measure of the relative surface error was used.

These results confirm, through quantitative measurements, important points about soft tissue modeling algorithms. First, if the underlying model is not appropriate, it is impossible to capture the deformation of the reference model, no matter the choice of parameters. This is well illustrated with the case of the linear elastic FEM model which cannot handle large displacements. On the other hand, our examples also show that it is possible to obtain rather good approximations of a given behavior using different methods (mass-spring model, co-rotational FEM, beam model) all within a range of computation times compatible with interactive simulations. We can also see that even an ideal, analytical model will not give the exact same result as an experiment, some of these differences coming from errors on the various measurements done on the experimental model. Our preliminary results also show the need for a variety of reference models, able to characterize various aspects of soft tissues, to clearly determine which algorithm is best for representing a particular behavior (linear elastic, visco-elastic, bi-phasic, porous, etc...). Similarly, it is important to define metrics that are most relevant to which property of an algorithm we want to evaluate. We have proposed an initial set of metrics to assess the accuracy of the models through comparisons with two different types of reference models. Additional metrics could certainly be proposed, in particular to evaluate the computational efficiency of the algorithms, which is also an essential element of interactive simulations.

Although the set of metrics and reference models presented in this work is limited, we believe they illustrate well the importance to quantitatively assess algorithms used in medical simulation. However, the main novelty of our approach lies in the combination of a unified, open framework where all models could

be compared, new metrics defined, algorithms and reference models added. This will eventually enable an unbiased comparison of the performance and accuracy of many different algorithms, to create an Open Benchmark for medical simulation.

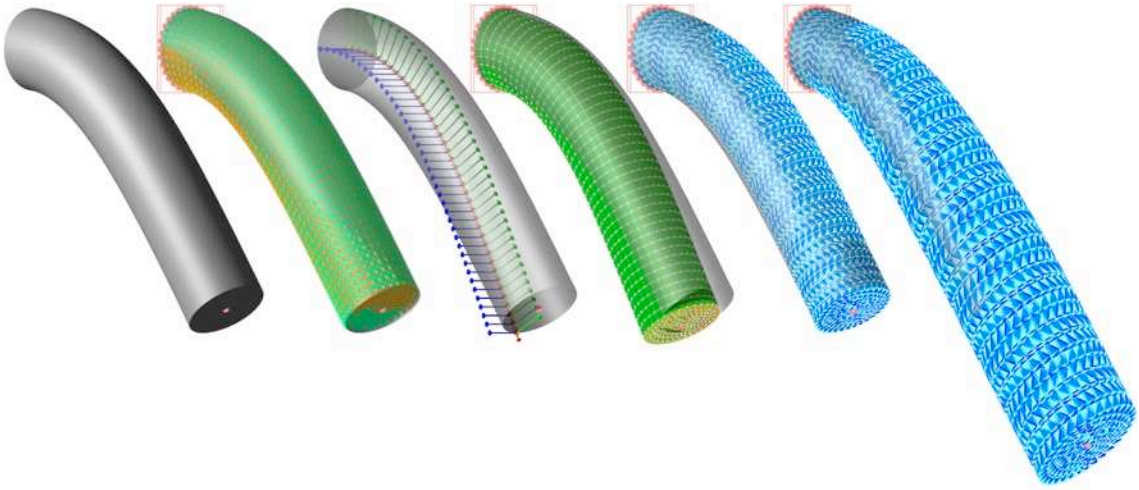


***Figure 53**: Simulation results for different real-time deformable models. From left to right: the biomechanical phantom, used as a reference, a mass-spring network model, a series of serially-linked beam elements, a FEM Hexahedral model, a co-rotational Tetrahedral FEM model, and a linear tetrahedral FEM model. The mean surface errors between each of these models and the reference phantom are, respectively: 0.75 mm, 4.32 mm, 2.87 mm, 0.63 mm, 18.6 mm.*

6. Performance metrics

It is universally accepted in critical-mission domains such as aviation that simulation promotes training while reducing risk. A growing body of empirical evidence supports these hypotheses for medical training. For instance, simulation-based training has been shown to accelerate metrics of proficiency for resident anesthesiologists (Abrahamson *et al.*, 1969) improve performance of surgical residents in laparoscopic cholecystectomy (Seymour *et al.*, 2002) and accelerate learning and enhance retention in medical students (Gordon *et al.*, 2006). Medical simulation training systems should provide the opportunity to learn and practice basic or advanced skills specific to a particular procedure (laparoscopy, interventional radiology, etc.). These skills can later be applied in the context of knowledge, judgment, and experience to ensure the safe conduct of medical procedures. An essential principle of educational theory is that learning be accompanied by evaluation both for formative (feedback) and summative (final assessment) purposes. The feedback can be used to create directed learning programs to enhance skill specifically in the areas that are deficient, then to verify that the criterion skill level has been attained. Yet, most academic or commercial simulators do not integrate performance metrics or other forms of assessment of the simulator as a training system.

Face Validity

Face validity pertains to whether the test “looks valid” to the examinees who take it. In medical simulation, it relates to the realism of a task or procedure in the simulator compared to the same task or

procedure as it would be performed in the operating room. While skills trainers can rely on a certain level of abstraction, procedural simulators, on the other hand, need to be visually (and haptically) realistic.

Construct Validity

Demonstration of construct validity is a process whereby the score in the training system is compared with a construct that can be reasonably assumed to correlate with technical skill in the operating room. This can be illustrated by showing that there are significant differences in the score given by the simulator to people that would be expected to differ in their laparoscopic skills (junior resident vs. intermediate level residents vs. chief residents).

Concurrent Validity

Concurrent validity refers to the relationship between concurrent measurements of the test score and another construct that reflects the measure being evaluated (technical skill). Significant correlation is expected to be found between scores obtained with the training system and the actual level of experience of the trainee.

Predictive Validity

In an ideal simulator, scores measured would be able to predict performance in the operating room. This is the best proof of validity of the simulator's metrics. In order for predictive validity to be tested, it is first necessary to develop a means to measure laparoscopic skills in the operating room. Predictive validity is difficult to assess as it is difficult to measure performance in the operating room in a quantitative way. The study performed by Seymour *et al.* (Seymour *et al.*, 2002) illustrated that the use of surgical simulation to reach specific target criteria significantly improved the OR performance of residents during laparoscopic cholecystectomy.

Performance metrics for laparoscopic surgery simulation

Metrics are widely employed in virtual environments and provide a yardstick for performance measurement. The current method of defining metrics for medical simulation remains more an art than a science. Through a collaboration with the Department of Surgery at Massachusetts General Hospital, we developed a new approach to defining metrics, specifically aimed at computer-assisted laparoscopic skills training and proposed a standardized global scoring system usable across different laparoscopic trainers and tasks (Cotin *et al.*, 2002) (Stylopoulos *et al.*, 2003) (Stylopoulos *et al.*, 2004). The metrics are defined in an explicit way based on the relevant skills that a laparoscopic surgeon should master. They rely on information provided by a five degree of freedom tracking device. These metrics have been integrated into a system capable of 1) simultaneously tracking the motion of two laparoscopic instruments, and 2) processing data from the instruments in real-time to provide immediate feedback and score about the task being performed.

The proposed metrics illustrate the relationship between the kinematic properties of the motion of laparoscopic instruments and the special cognitive and psychomotor skills that a laparoscopic surgeon should master. Additionally, a standardized method of performance assessment is imperative to ensure comparability across different systems and training conditions. In this section we first introduce the concept of task-independent metrics for skills assessment and how it relies on kinematics analysis. We

describe the various parameters used to represent the essential characteristics of any laparoscopic task, and how these parameters are combined into a standardized score.

Standardized performance assessment

It is not clear how surgeons learn and adapt to the unusual perceptual motor relationships in minimally invasive surgery. A major part of the learning process, however, relies on an apprenticeship model according to which an expert surgeon qualitatively assesses the performance of the “novice”. In order to define a quantitative performance metric that is useful across a large variety of tasks, we looked at the way expert surgeons instruct and comment upon the performance of novices in the operating room. Expert surgeons are able to evaluate the performance of a novice by watching on the operating room monitor the motion of the visible part of the instruments that have been introduced into the abdominal cavity. Based on this information and the outcome of the surgical task, the expert surgeon can characterize qualitatively the overall performance of the novice on each of the key parameters that are required for efficient laparoscopic manipulations.

We identified the following components of a task that account for competence while relying only on instrument motion: compact spatial distribution of the tip of the instrument, smooth motion, good depth perception, response orientation, and ambidexterity. Time to perform the task as well as outcome of the task are two other important aspects of the ”success” of a task that we decided to include. Finally, in order to transform these parameters into quantitative metrics, we relied on kinematics analysis theory that has been successfully used in previous work to study psychomotor skills (Mavrogiorgou *et al.*, 2001). Most laparoscopic tracking devices or haptic interfaces can provide information about kinematic parameters, in particular: position of a three-dimensional point representing the tip of the instrument, rotation of the instrument about its axis, and degree of opening of the handle. All these variables are time-dependent and are represented by the following quantities: $[x(t),\ y(t),\ z(t)]^{\mathrm{T}}$ is the three-dimensional position of the tip of the instrument, and $\theta(t)$ is the rotation of the instrument about its axis. Grasper opening is not taken into account in our metrics. The five kinematic parameters we have defined are:

- **Time**: this is the total time required to perform the task (whether the task was successful or not). It is measured in seconds and represented as P_1 with $P_1 = T$.
- **Path Length**: also called economy of motion, it corresponds to the length of the curve described by the tip of the instrument over time. In several tasks, this parameter describes the spatial distribution of the tip of the laparoscopic instrument in the workspace of the task. A compact ”distribution” is characteristic of an expert. It is measured in centimeters and represented as P_2

$$P_2 = \int_0^T \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2}\, dt \quad (69)$$

- **Motion Smoothness**: this parameter is based on the measure of the instantaneous jerk defined as $j=d^3x/dt^3$ and represents a change of acceleration and is measured in cm/s^3 . We derive a measure of the integrated squared jerk J from j as follows

$$J = \sqrt{\frac{1}{2}\int_0^T j^2 dt} \quad (70)$$

The time-integrated squared jerk is minimal in smooth movements. Because jerk varies with the duration of the task, J has to be normalized for different tasks durations. This was done by dividing J by the duration T of the task: $P_3 = J/T$.

- **Depth Perception**: we measure depth perception as the total distance travelled by the instrument along its axis. It is represented as P_4 and can easily be derived from P_2 .

- **Response orientation**: this parameter characterizes the amount of rotation about the axis of the instrument and illustrates the ability to place the tool in the proper orientation in tasks involving grasping, clipping or cutting. It is represented as P_5 and measured in radians

$$P_5 = \sqrt{\int_0^T \frac{d\theta^2}{dt} dt} \quad (71)$$

All of these parameters can be seen as cost functions where a lower value describes a better performance. Task-independence is achieved by computing the z-score (Howell, 1998) of each parameter P_i. The z-score z_i corresponding to parameter P_i is defined as follows

$$z_i = \frac{P_i^N - \bar{P}_i^E}{\sigma_i^E} \quad (72)$$

where $\bar{P}_i^E$ is the mean of $\{P_i\}$ and σ_i^E is the standard deviation for the expert group. P_i^N corresponds to the result obtained by the novice for the same parameter. Assuming a normal distribution, 95% of the expert group should have a z-score $z_i \in [-2; 2]$. Therefore we can limit the range of values for z_i to $[-z_{max} ; z_{max}]$ with $z_{max} > 2$. A standardized score is computed from the independent z-scores z_i according to the following equation

$$z = 1 - \frac{\sum_{i=1}^{N} \alpha_i z_i}{\sum_{i=1}^{N} \alpha_i z_{\max}} - \alpha_0 z_0 \quad (73)$$

where N is the number of parameters, z_0 is a measure of the outcome of the task and α_0 the weight associated with z_0. There are various ways of evaluating z_0 and it can be either a binary measure ($z_0 = 0$ for success, $z_0 = 1$ for failure), or a more complex measure, as introduced by (Rosser *et al.*, 1998) for instance. Similarly, α_i is the coefficient for a particular parameter P_i.

Three tasks of increasing difficulty selected from established training programs (the Yale Laparoscopic Skills and Suturing Program, the SAGES-FLS training program and the graded exercises used at Harvard Center of Minimally Invasive Surgery) were examined:

- **Task 1 - Peg Board Transfer**: the purpose of this task is to assess eye-hand coordination, bimanual dexterity, and depth perception,
- **Task 2 - Cobra Rope Drill**: the purpose of this task is to assess depth perception, two-handed choreography, and non-dominant hand development,

- **Task 3 - Needle-Cap**: the purpose of this task is to assess depth perception, non-dominant hand development, and targeting.

To validate our scoring system, we designed the CELTS system (Computer-Enhanced Laparoscopic Training System) and conducted a study comparing experts and novices surgeons. The expert group consisted of staff surgeons from our hospital, while the novice group consisted of 20 surgeons in training. Each of the experts was asked to perform each task several times. Each of the novices was asked to perform each task once, and was visually assessed by an expert. Most of the novices had had prior exposure to the tasks they were asked to perform. The values of the different parameters $\{P_i\}$ as well as the score z were computed using our software platform and recorded on file. The results of the study illustrate several important aspects of our method. First, they confirm that our metrics are independent of the task being performed. Without changing any of the parameters P_i or weights α_i used in the computation of the overall score, our method still provides an efficient way of discriminating between expert and novice performance, irrespective of the task, as illustrated in the table below.

	Task 1	**Task 2**	**Task 3**
Expert mean score (standard deviation)	1.0 (0.035)	1.0 (0.03)	1.0 (0.015)
Novice minimum score	0.22	0.38	0.04
Novice maximum score	0.69	0.58	0.72

***Table 4**: This table presents the overall scores obtained by novices and experts in our study. It clearly highlights the gap existing between expert's and trainee's performance, irrespective of the task. The best score obtained in the novice group was more than 2 standard deviations away from the mean score of the expert group.*

In addition to an overall measure that discriminates between experts and non-experts, our method provides additional feedback by identifying the key factors that contribute to the overall score. For instance, we can see in Figure 54 that factors such as depth perception, smoothness of motion, and response orientation are major indicators of performance, while time and path length (often referenced in the literature) do not provide enough information to capture the magnitude of the difference. By comparing the expert's verbal assessment and the parameters P_i for which existed a significant difference between novice and expert, a high correlation was found, thus validating our choice of parameters.

A second validation study was setup to compare CELTS to other means of training, including two other simulators. CELTS was found to provide valuable, quantitative and qualitative feedback. In particular the scoring system we proposed was evaluated and has been tested head to head with two other medical simulation systems and found to be statistically significantly superior to both (Maithel, *et al.*, 2006). Subjects were voluntarily enrolled at the Learning Center during the 2004 SAGES annual meeting. Each subject completed two repetitions of a single task on each of three simulators, MIST-VR, Endotower, and CELTS; performance scores were automatically generated and recorded. Scores of individuals with various levels of experience were compared to determine construct validity for each simulator. Experience was defined according to four parameters: (a) PGY level, (b) fellowship training, (c) basic laparoscopic cases, and (d) advanced laparoscopic cases. Subjects rated each simulator regarding six face validity (realism of simulation) parameters using a 10-point Likert scale (10 = best rating) and participant scores were compared to previously established expert scores (proficiency goals for training). Nearly 100 attendees completed the study. Construct validity was demonstrated for all three simulators; significant

differences in scores were detected according to one parameter for MIST-VR, two parameters for Endotower, and all four parameters for CELTS. Construct and face validity were rated as good to excellent for all three simulators but the scoring system is CELTS was demonstrated to be a lot more accurate than in the other two systems (experts obtained the highest scores, novices the lowest scores and interns with intermediate level of training obtained average scores).

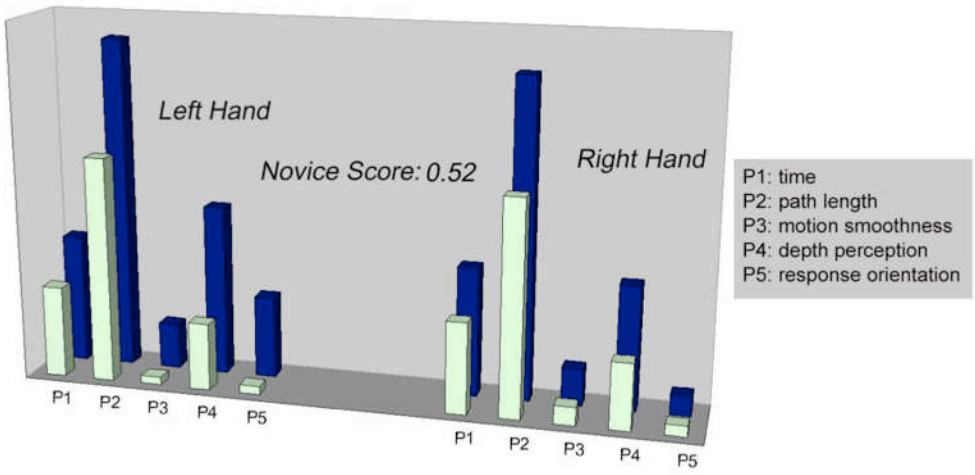


***Figure 54**: This diagram illustrates the comparison between the experts group (light) and a novice (dark). The overall score given to the novice by our system was 0.52; this score can be explained by low values associated with particular parameters (P3, P4, P5). Such feedback is very important to provide a meaningful interpretation of the score.*

The conclusion of these different studies illustrate the novelty of the CELTS system as a standardized skills trainer that combines the advantages of computer simulation with the features of the traditional and popular training boxes. More importantly these studies validate our choice of metrics, their relevance and robustness, allowing CELTS to be used with a wide array of tasks and across different training conditions.

Summary of contributions

- **Research articles**
  - S. Cotin, N. Stylopoulos, M. Ottensmeyer, P. Neumann, D. Rattner, and S. Dawson. “Metrics for Laparoscopic Skills Trainers : The Weakest Link!”. In Proceedings of MICCAI 2002, Lecture Notes in Computer Science 2488, p. 35-43, Springer-Verlag, 2002.
  - N. Stylopoulos, S. Cotin, S. Maithel, M. Ottensmeyer, P. Jackson, Bardsley, P. Neumann, D. Rattner, and S. Dawson. “Computer-enhanced laparoscopic training system (CELTS): bridging the gap”. Surg Endosc. 18 (5):782-9, 2004.
  - N. Stylopoulos, S. Cotin, S. Dawson, M. Ottensmeyer, P. Neumann, R. Bardsley, M. Russell, P. Jackson, and D. Rattner. “CELTS: A clinically-based computer enhanced laparoscopic training system:. Proceedings of 11th Annual Meeting, Medicine Meets Virtual Reality, pp. 336-342, 2003.
  - A. Kerdok, S. Cotin, M. Ottensmeyer, A. Galea, R. Howe, and S. Dawson. “Truth Cube: Establishing Physical Standards for Real Time Soft Tissue Simulation”. Medical Image Analysis; 7:283-291, 2003.
  - D. Valtorta, M. Hollenstein, A. Nava, V. Luboz, M. Lu, A. Choi, E. Mazza, Y. Zheng, and S. Cotin. “Mechanical characterization of soft tissue: comparison of different experimental techniques on synthetic materials”. Proceedings of the Fourth International Conference on the Ultrasonic Measurement and Imaging of Tissue Elasticity, p 94, 2005.
  - V. Luboz, X. Wu, K. Krissian, C.-F. Westin, R. Kikinis, S. Cotin, and S. Dawson. “A segmentation and reconstruction technique for 3D vascular structures”. Proceedings of the MICCAI Conference, pp. 43-50, 2005.
  - M. Marchal, E. Promayon, and J. Troccaz. “Simulating Prostate Surgical Procedures with a Discrete Soft Tissue Model”. Third Eurographics Workshop in Virtual Reality, Interactions, and Physical Simulations (VriPhys'06), pp. 109-118, 2006.
- **Software**
  - **SOFA**: validation metrics for the assessment of deformable models were integrated within the SOFA framework.
  - **CELTS - Computer Enhanced Laparoscopic Training System**: implementation of a new concept for performance metrics and scoring based on kinematics analysis; development of a fully functional prototype that has passed several validation tests, and was ranked first in a validation study involving several existing commercial products. Development time: 18 months; unique developer; about 5,000 lines of code.
- **Patent**
  - Cotin S, Stylopoulos N, Dawson S, Ottensmeyer M inventors; “Surgical Training System for Laparoscopic Procedures”, United States Patent Application, 10/797,874 – March 10, 2004.
- **Miscellaneous**
  - **Truth Cube:** all the raw and processed data concerning the Truth Cube was made publicly available on a website: http://biorobotics.harvard.edu/truthcube
  - **CELTS**: several discussion about the commercialization of CELTS took place from 2004 to 2006.

Chapter V
Systems Integration & Prototypes

1. Introduction

A constant motivation in our research has been to work as closely as possible with physicians, in an effort to understand their needs, but also to evaluate our results. While the first point is an obvious requirement in this research field, the second requires to develop prototypes that can be tested by clinicians. This demands a large additional effort compared to simply describing our results in conference or journal articles. Yet, it is through the development of such prototypes that we can change the vision physicians have of medical simulation, and eventually make medicine safer.

Developing a complete simulator, prototype, or even a simpler proof of concept is a very demanding task. In addition to the development of new algorithms, a major difficulty in the development of a simulation system comes from the integration of all the necessary components to provide a meaningful prototype which can then be evaluated by clinicians. This integration task requires large developments efforts, but it is also challenging since each component of a simulator needs to run at a specific frequency and exchange data with several other components in the system. To ease the construction of simulators, the creation of an efficient and flexible software framework has been an increasingly active topic in the medical simulation community in recent years. Such a framework would need to provide a run-time kernel template and a generic programming interface for all components. We will see in the following chapter (Chapter VI) an introduction to SOFA, the Simulation Open Framework Architecture, an initiative we have started a few years ago. The main objective of SOFA framework is to simplify the development and integration of new simulation components, thus accelerating the development of prototypes. With the various computational constraints of interactive simulation, not only different algorithms need to be integrated within a system, but also different hardware architectures need to be supported. With the recent development of GPU-based methods, it is essential that new generations of simulation frameworks allow flexible integration of multi-core CPUs and GPUs.

The simulators that are presented in the following sections materialize an important part of the research we have done during nearly ten years. They address specific training requirements in the areas of emergency medicine, laparoscopic surgery, interventional radiology, and ophthalmology. These different prototypes of simulators address various needs, with varying degrees of completion. Some are just proofs of concept, while others are closer to prototypes. One even became a product. They also emphasize different aspects of our research, from modeling soft tissues to performance metrics, advanced rendering or new collision response methods. Finally, and more importantly, designing and creating these prototypes has been an essential element of communication with physicians to better understand their needs.

Emergency medicine

Emergency medicine and anesthesiology have been the first clinical targets of simulation systems. The first systems to be introduced in this area were patient simulators, i.e. mannequins equipped with various sensors and basic hemodynamics functionalities. These systems have now become a *de facto* standard for training in emergency medicine as they have become more reliable and include more functionalities. They are very often connected to a computer that controls some of the parameters and responses of the system. Yet, an instructor is often required to control how the mannequin will react according to the actions of the trainee. One of the characteristics of emergency medicine simulation is that it shares several characteristics with open surgery, for which tracking and haptic interfaces are not readily available, thus requiring to use other approaches.

Chest Trauma Training System

From 2001 to 2003 we developed a chest trauma treatment training system (VIRGIL™) which integrates a realistic physical mannequin, simulated tissues for haptic response, mathematical models of tissue tool interactions, and augmented reality to produce a record of performance during treatment of unilateral or bilateral chest trauma. Initially designed for army medic training in realistic outdoor situations, a classroom version was later designed (see Figure 55) to provide basic training for civilian emergency medicine personnel.

Figure 55: *Prototype of the VIRGIL chest trauma training system. This version is the "classroom" version, another prototype was designed for use during military exercises. Both systems use tracking devices for the instruments, a touch screen for user interface navigation, and augmented reality to present the outcome of the training session.*

VIRGIL™ was the first training system developed with the Sim Group at CIMIT (Center for Integration of Medicine & Innovative Technology). CIMIT's mission is to apply advanced technology to medicine in effective ways. As part of this effort, we worked closely with the US Army Medical Research & Materiel Command and the Special Forces Medical Command to create a new method to train medics for combat casualty care. The VIRGIL™ system combines the use of a realistic mannequin, an integrated computer system, a touch-based graphical interface, and electro-magnetic position trackers that follow the motion

of various instruments (chest dart, chest tube, syringe) during training exercises. VIRGIL™ provides realistic force feedback during the skin incision, dissection through intercostal muscle and pleura, and subsequent placement of a 36 Fr chest tube. As the educational scenarios become progressively more difficult, the system tracks the trainee's progress and detects patterns of error. The system is used in a trainee/instructor configuration, with about ten minutes required per trainee. A web-based educational tutorial is also available for refresher training.

VIRGIL™ was the first practical step of a long-term research program we developed at CIMIT, and directly addressed the expressed needs of the Special Forces Medics to learn and practice safe treatment of combat chest trauma. The simulator combines sophisticated 3-D anatomic models generated from CT Scans of actual human anatomy with a mannequin built utilizing the same measurements as the computer models. Since the internal organs are proper in size, location, and density, mistakes that would happen in the real world will also happen realistically in the simulator. This anatomic realism contributes to "transfer of learning" from the simulated world to real world trauma scenarios.

Through a collaboration between the Uniformed Services University of the Health Sciences (Bethesda, MD) and the Boston Med-Flight, the Sim Group participated in a series of controlled studies designed to validate the VIRGIL™ Chest Trauma Training System in a classroom environment. The study demonstrated that computer enhanced training, using augmented reality, is more effective than training on animals for teaching chest tube insertion in a military medical environment (Bowyer, 2006). Participants responded enthusiastically to VIRGIL™ citing better visualization and increased understanding of the procedure. These different results, and our ability to answer practical needs through an innovative approach, were the reasons why the VIRGIL project was awarded the Top Ten Army's Greatest Inventions Award in 2003.

2. Laparoscopic Surgery

Computer-based medical simulation has known an important development with the widespread introduction of laparoscopic surgery in Medicine. The rationale for this increased interest came from the need to train surgeons (novice or experienced) to this new technique. While it is theoretically possible to reproduce the different aspects of a laparoscopic procedure on a computer, in practice many challenges remain to be solved: real-time simulation of the deformation of anatomical structures, simulation of cutting, dissection, suturing and other tissue manipulations that modify the topology of the deformable model, collision detection and contact response to describe tissue-tool interactions, or haptic feedback. To enable interactivity of the simulation, all the components of the simulation are constrained by the need for real-time computation, therefore creating additional requirements for the integration of these different components. Finally, designing a training system requires to add educational content to the simulator, as well as means to assess the performance of the trainee. In the following paragraphs, we describe two systems for laparoscopic training. The first one was develop during my PhD and integrates all aspects of the simulator, including haptic feedback. The second one specifically addresses the issue of performance assessment, using metrics described in Chapter IV.

Laparoscopic simulation for hepatic surgery

To evaluate the potential of the work done during my Ph.D. on deformable models I developed a prototype of simulator for laparoscopic surgery, using a dedicated haptic interface. This prototype integrates the different features described previously (anatomical model, biomechanical models, and speed-up algorithm) as well as a collision detection algorithm, and a force feedback controller (Figure 56). The components run on a distributed architecture composed of a PC and a Digital Equipment Computer Alpha Station. The force feedback device (Laparoscopic Impulse Engine) is connected to the PC (Pentium 166 MHz) for technical reasons.

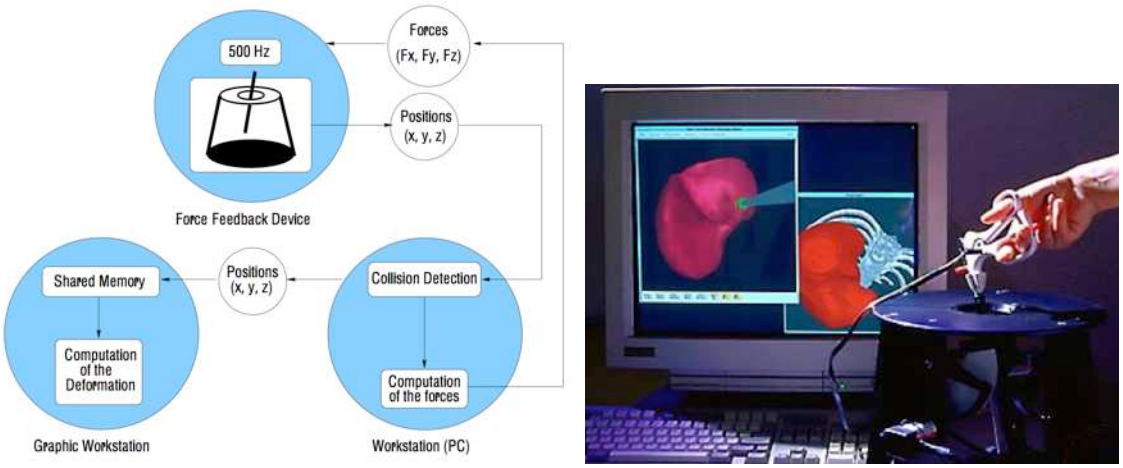


***Figure 56**: the surgeon manipulates the force feedback device. If a collision is detected with the surface of the virtual organ, the mesh is deformed in real-time and a non-linear reaction force is computed and sent back to the force feedback system..*

The collision detection is also performed on the PC as well as the force evaluation since it is possible, thanks to our pre-processing algorithm, to split the computation of the forces and the computation of the deformation. The deformation is performed on the Alpha Station (400 MHz with 3D graphics hardware) as well as the display of the different parts of the scene (see Figure 56) and the communication between the two machines is performed via an ethernet connection. The data transmitted between the two computers is limited to the five degrees of freedom of the laparoscopic instrument plus some information related to collision detection. Consequently, we have a very high frequency (> 300 Hz) in the simulation loop with very little latency between force feedback and visual feedback.

This first prototype was further developed in subsequent Ph.D. thesis, with a series of improvements essentially focused on tissue-tool interactions. Yet, this initial system was efficient enough to demonstrate our work at several scientific or clinical meetings. It also led to a publication in Annals of Surgery that received great attention from the medical community.

Laparoscopic skills training

In 2003 we started the development of an unconventional laparoscopic training system, essentially designed to validate our ideas on performance metrics. The system, called CELTS (for Computer

Enhanced Laparoscopic Training System) uses kinematics-based performance metrics to provide accurate, instructor-independent, feedback to the trainee. The system can access wirelessly a database of trainee and expert performance records, to store new score and detailed training information. This database can be accessed through a web portal by an instructor to monitor the progress of one or several surgical residents. CELTS also presents a variety of interchangeable tasks, from simple ones such as reaching objects using laparoscopic graspers to complex suturing tasks. The CELTS system received great interest from the clinical community for its practicality and the needs it addressed. One advantage of CELTS over computer-based virtual reality systems is that it allows the trainee to use whatever instruments he or she uses in the operating room (see Figure 57). It provides haptic feedback that is absent from most computer-based systems as well as complex tasks, such as suturing, that are still primitive in virtual environments. However, to be an effective educational tool, the metrics associated with the simulator must provide meaningful information to the student. To be used for evaluation, the data also needs to be demonstrated reliable and valid. Thus, CELTS has gone through a rigorous process of validity testing.

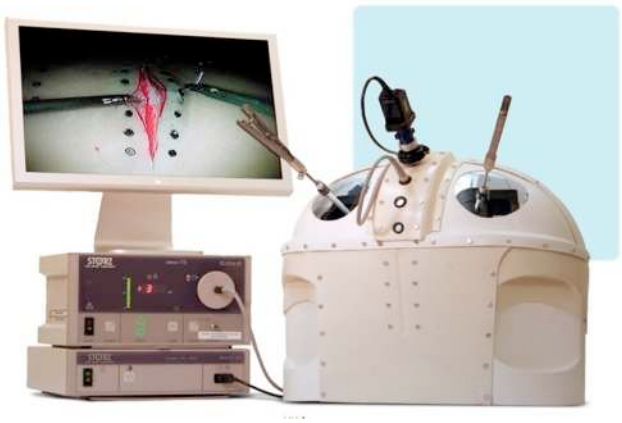


Figure 57: *Prototype of the Computer Enhanced Laparoscopic Training System (CELTS). The system was initially designed to validate a new concept for performance metric in laparoscopic surgery but evolved rapidly to a fully autonomous training system. It was used for several studies and in daily practice at Massachusetts General Hospital.*

The CELTS system relies on a compact and modular design. The core of the system (see Figure 57) is mostly a shell and a tracking device that can accommodate any type of laparoscopic device. The camera, light source and other laparoscopic equipment can be easily connected to the system. The tracking interface is used to measure the motion of the tip of the laparoscopic instruments. It uses a modified Virtual Laparoscopic Interface[5]). The VLI includes tool handles similar to laparoscopic instruments, but there are no tool tips with which to manipulate real objects. To permit the use of the VLI as a tracking instrument, the main shafts and tool handles from the VLI were removed, and replaced with a system which allows the use of a variety of laparoscopic instruments. By replacing the main shaft of the VLI and modifying the laparoscopic instruments, the pitch/yaw/roll and thrust sensors can be used without modification. The gripper sensor in the original system is a Hall effect sensor that detects the motion of the actuator shaft in the tool handle. The instrument modifications and Hall effect sensor placement were performed as precisely as possible, but due to the sensitivity of the sensor to small changes in the distance between it and the magnet, each instrument was separately calibrated. This ensures good

[5] VLI, Immersion Corp., San Jose, CA.

correspondence between the true gripper axis motion and that measured by the system. Changing the calibration constants and offsets when switching tools is computed by software. For visual feedback we used a fixed surgical endoscope, camera, and light source (Telecam SL NTSC/Xenon 175, Karl Storz Endoscopy-America, Inc., Culver City, CA), to provide the same visual feedback encountered in minimally invasive procedures.

We developed a software interface that integrates data processing as well as visualization of the instruments motion, instruments path, normalized score. Details about the scoring system are provided in Chapter IV. Our system uses the Virtual Laparoscopic Interface API (Immersion Corp.) as a basis for communication with the VLI. The raw data consists of time-stamped values of the position and orientation of each of the two laparoscopic instruments, recorded at a sampling rate of about 20 ms. Before computing any of the kinematic parameters, the raw data is filtered. We have implemented various low-pass filters, using mean and median filtering. To compute accurately high order derivatives of the position, we used a second-order central difference method (if no filtering and/or first order methods were used, an accurate computation of the jerk would be very difficult). We implemented the user interface using C++, FLTK5 , and OpenGL. The user interface offers real-time display of the tip of the tool, and its path. Kinematics analysis and computation of the score are performed at the end of the task, providing immediate information to the user. Moreover, a visual comparison of the results of the experts group vs. the novice group illustrates clearly what skills need to be improved in order to get a higher score.

Figure 58: *Web interface for the CELTS system. All score computed during a training session, as well as detailed metrics, are exported to a database which is remotely accessible through a web interface.*

Two validation studies were performed to evaluate CELTS i.e. to assess face and construct validity, quality of the haptic feedback, flexibility of the design, accuracy of the scoring system. The second validation study was performed by an independent group, and compared CELTS to two other training systems. Construct and face validity were rated as good to excellent for all three simulators but the scoring system is CELTS was demonstrated to be a lot more accurate than in the other two systems (experts obtained the highest scores, novices the lowest scores and interns with intermediate level of training obtained average scores).

Although full computer-based simulators hold great promise to expand the scope of laparoscopic simulation, current haptic interfaces may limit their utility for assessment. Computer-enhanced skills trainers are for the time being a very interesting alternative, offering an intuitive interface while incorporating relevant metrics.

3. Interventional Radiology

Over the last twenty years, interventional methods such as angioplasty, stenting, and catheter-based drug delivery have substantially improved the outcomes for patients with vascular disease. Pathologies that used to require a surgical procedure can now be treated in a much less invasive way. As a consequence, interventional radiology procedures represent an increasing part of the interventions currently performed, with more than a million patients treated this way in Europe or the United States. However, these techniques require an intricate combination of tactile and visual feedback, and extensive training periods to attain competency. To reinforce the need for reaching or maintaining proficiency, the FDA recently required that US physicians go through simulation-based training before using newly developed carotid stents.

Vascular diseases are the number one cause of death worldwide, with cardiovascular disease alone claiming an estimated 17.5 millions deaths in 2005. An increasingly promising therapy for treating vascular diseases is interventional radiology, where a guidewire-catheter combination is advanced under fluoroscopic guidance through the arterial system, thus allowing a more localized therapy while reducing recovery time for the patient when compared to surgical procedures (Figure 59). However, the main difficulty in interventional radiology therapies comes from the difficulty to navigate within complex vascular networks while relying on two-dimensional X-ray views of the patient. Yet, the best training method so far has been actual interventions on patients with a vascular pathology.

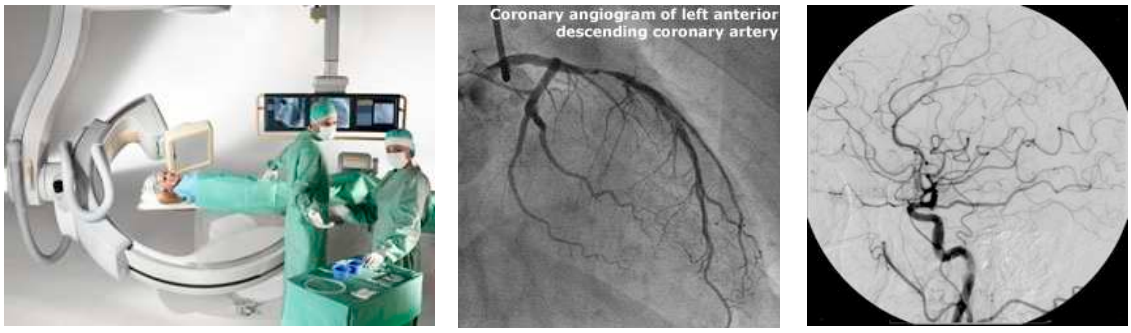


Figure 59: *Interventional radiology. (Left): operating room with C-arm, operating table, and monitors; (Center): an example of coronary angiogram, and (Right): an example of cerebral angiogram.*

Interventional Cardiology

Interventional Cardiology is a branch of cardiology that deals specifically with catheter-based treatments of heart diseases. Coronary angioplasty and coronary stenting are procedures performed to open up a blocked coronary artery and restore blood flow to the heart muscle. Often used as an alternative treatment to coronary artery bypass surgery, they are less invasive, although technically challenging. Angioplasty consists in navigating a balloon-tipped catheter at the site of the blockage, and then inflating the balloon to expand the diameter of the vessel. In most cases, following balloon angioplasty, a stent will also be placed to keep the artery open.

ICTS: Interventional Cardiology Training System

The work described in this section was done from 1999 to 2000 while I was working with the Medical Application Group at Mitsubishi Electric in collaboration with CIMIT and the Massachusetts General Hospital in Boston. During this period we developed a computer simulation of interventional cardiology catheterization named ICTS. This simulation integrated clinical expertise, research in learning, and technical innovations to create a realistic simulated environment. The goal of this training system is to augment the training of new cardiology fellows as well as to introduce cardiologists to new devices and procedures. To achieve this goal, both the technical components and the educational content of the ICTS brought new and unique features: a simulated fluoroscope, a physics model of a catheter, a haptic interface, a fluid flow simulation combined with a hemodynamic model and a learning system integrated in a user interface (Figure 60). The simulator was able to generate (in real-time) high quality X-ray images from a 3D anatomical model of the thorax, including a beating heart and animated lungs. The heart and lung motion were controlled by the hemodynamic model, which also computed blood pressure and EKG. Vascular devices, such as catheters, guidewires or angioplasty balloons were represented using a multi-body dynamics model. A haptic device was designed to control the specific devices of a interventional radiology procedures and provided appropriate feedback when contact with a vessel wall was detected.

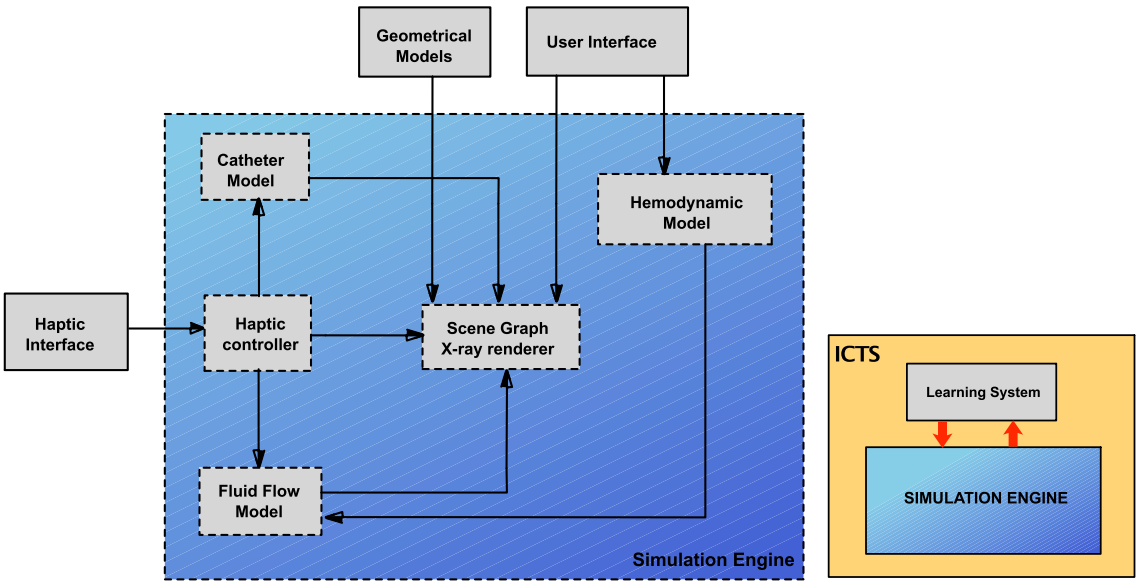


***Figure 60**: Simulation engine and learning system for the ICTS, It is the combination of an efficient simulation engine and a intuitive learning system that defines a complete training system.*

Once the catheter is in place, a contrast agent can be injected into the coronary arteries; blood and contrast agent mixing is computed and a visual representation of the angiogram is displayed by the X-ray renderer. By bringing key advances in the area of medical simulation and by integrating in a single system both a simulation engine and learning tools, the ICTS opened new perspectives for computer-based training systems.

Technical overview

The following paragraphs briefly describe the key characteristics of the different components of the systems, as well as their integration within a prototype.

Anatomical modeling

Geometric modeling is used to provide information about the thoracic anatomy and the attenuation coefficients which will be used to generate the x-ray images. The anatomical models used in the ICTS can be divided in three classes: polygonal representations of the static anatomy based on segmented data from the Visible Human Project; animated polygonal models, i.e. the heart and lungs, created in Maya®, an animation and modeling package; and animated NURBS surfaces (correlated to the cardiac motion) to model. The geometrical and topological characteristics of the vascular system have been created in collaboration with a cardiologist to ensure the accuracy of the representation. Finally, an x-ray attenuation value is associated with each anatomical model. To create a beating heart, the heart model and arteries have been key-framed while a deformation was applied to the polygonal and NURBS surfaces. The result is a cyclic regular heartbeat, exhibiting all the characteristics of a real heart (twist, elongation, contraction). Moreover, control of the heart rate is possible from the user interface.

Hemodynamics

The hemodynamics component provides simulated patient physiology. In particular, it provides information related to coronary blood flow. Two categories of hemodynamic parameters are computed: steady state and transient characteristics. The steady state hemodynamic characteristics of the patient describe parameters whose values are assumed to not change during any given cardiac cycle, for instance heart rate, vascular compliance, and vessel resistance. Transient characteristics of the circulation and heart action have values that change in a continuous way during the cardiac cycle, i.e. aortic root pressure, ECG values, and ventricular volumes. Steady state values are computed based on descriptions and mathematical models available from different sources. These steady state values provide “bounding boxes”, i.e. control parameters, that shape the transient components. The hemodynamic model does not provide a full control feedback. Rather, control is provided by an “instructor” who can, for instance, change the heart rate, at any time.

Fluid Flow

One-dimensional blood flow is computed using Poisseuille’s law (Guyton *et al.*, 1995). Since the vascular system of the heart contains only a few dozen vessels, the resistance of each vessel can be modified “on the fly” to simulate a stenosis. The driving pressures (aortic and myocardial) are provided by the hemodynamic module. As a result, we obtain a real-time pulsatile one-dimensional flow in the coronary arteries. When contrast is injected into the coronary arteries, mixing between blood and contrast agent is computed and defines the intravascular contrast density used to compute the visual properties of the vessels in the synthetic X-ray images.

Devices

The catheter model simulates flexible instruments, such as catheters or guidewires, as they interact and deform within the aorta and coronary arteries. In the ICTS such devices are represented as a multi-body

system, i.e. a set of rigid links connected by joints. The multi-body object can only be controlled by applying motion to the joints or forces and moments to the links and joints. Three different forces can be applied to the multi-body object: contact forces, injection forces and force applied by the user at the proximal end of the instrument. Contact forces are the result of the interaction of the tool with vessel walls and are computed using a penalty method.

Haptics

The haptic interface device used in the ICTS is a custom-designed passive force feedback system for catheter-like instruments (see Figure 62). Our haptic interface consists of a tracking device to measure catheter translation and rotation and independently controlled servomotors which produce force and torque resistance. Motion measurements are sent to the simulation and combined with other data from the physical model to compute the proximal catheter force and torque. In addition to the catheter motion measurement, the interface device senses the pressing of a momentary contact foot switch to replicate the on/off action of the fluoroscope as well as the motion of the syringe's plunger during injection of contrast.

Rendering

In interventional cardiology, visual feedback through fluoroscopically controlled contrast injection is the major source of information for the cardiologist. The method used for generating X-ray and fluoroscopic images in the ICTS is based on polygonal models associated with specific X-ray attenuation coefficients. This permits real-time realistic fluoroscopic image computation and rendering on OpenGL accelerated hardware. The three-dimensional polygonal models can be either reconstructed from medical images or generated using a 3D modeling software.

Systems integration

Simulation engine

A major difficulty in the creation of a simulator, besides the development of the different modules described above, is the integration of these modules in a real-time framework. Since each component of the simulator needs to run at a specific frequency and exchange data with several other components, the integration task is usually challenging. To achieve this goal, we used *Control Shell*, a component-based real-time programming tool for system development combined with a real-time control framework. The system, initially developed on a 4-processor Onyx 2 workstation was later implemented on a 4-processor PC workstation. With the integrated system, a higher order approach to procedural simulation is enabled, namely the transition from the simulation as a training system to the simulation as a means for learning. It is our firm conviction that successful medical simulation must permit more than mere technical repetition. It must impart knowledge which can be transferred from the simulation to direct patient care.

User Interface and Learning System

We embedded the simulation engine, i.e. the integrated, closed-loop system of modeling, rendering and haptics, in a larger context of "virtual rounds" (Shaffer *et al.*, 1999). Currently, in cardiac catheterization labs, trainees learn by examining a case history, deciding on a treatment protocol, performing procedures on actual patients, etc. The concept of virtual rounds is to

replicate these steps “virtually”, through the simulation. To implement the idea of virtual rounds, technical functionality of the simulation engine are used to support the clinical and pedagogical aspects of the training system, and the user interface becomes the enabling portal to the different areas of the training system (see Figure 61).

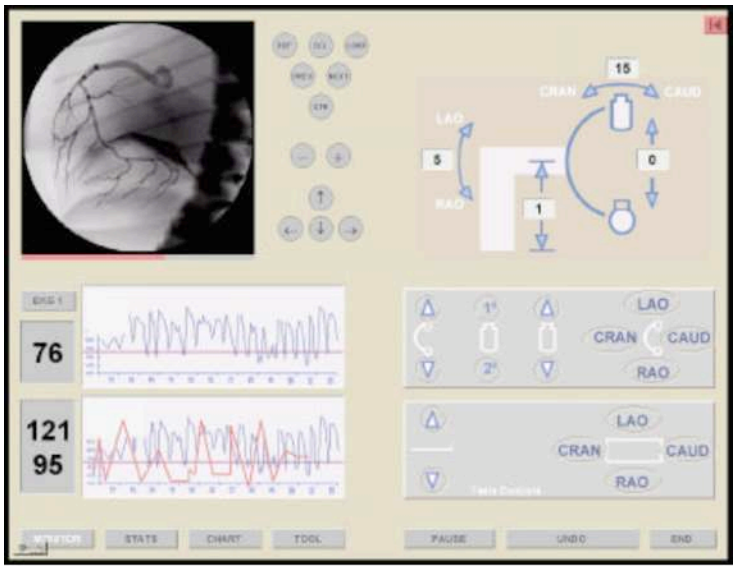


***Figure 61**: screenshot of the ICTS user interface. On this view, the fluoroscope controls and hemodynamic parameters are shown.*

Evolution

The prototype that was developed during the 18 months of this project is illustrated in Figure 62. It was successfully demonstrated at the American Heart Association annual meeting in 2000 where it was very well received.

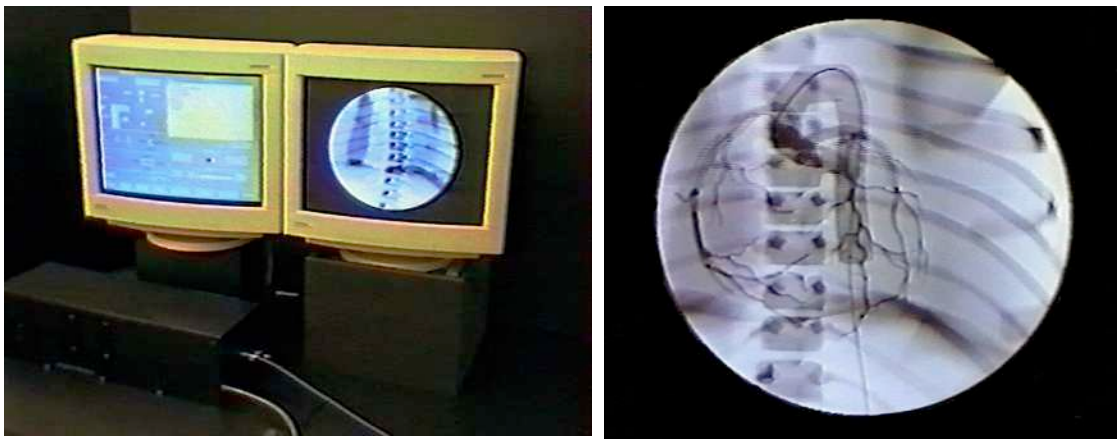


***Figure 62**: (Left) A picture of the ICTS prototype with the haptic device in the foreground. (Right) a screenshot of the simulated fluoroscope showing the catheter and contrast moving into the aorta and coronary arteries.*

The project was then transfered to a British company, Virtual Presence, that continued our work for some time before selling the new prototype to Mentice[6], which started to commercialize it under the name VIST in 2003 (see Figure 63). Since then, VIST has been used to train thousands of interventional radiologists, in cardiology but also in other specialties such as neuroradiology and interventional radiology of the peripheral vascular system.

***Figure 63**: view of the complete VIST system for interventional radiology training. Although a major redesign of the hardware components was performed as well as improvements of the software, the core and the basic principles of the ICTS still apply.*

Interventional Neuroradiology

Stroke, the clinical manifestation of cerebrovascular disease, is the third leading cause of death in the United States. According to the National Institute of Health, more than 700,000 strokes occur every year in the United States, resulting in over 200,000 deaths [REF]. In Europe, heart and stroke deaths are the main cause of mortality with 4.3 million deaths each year. The single most common cause is coronary heart disease with more than a million deaths, followed by stroke with approximately 650,000 deaths each year.

In the United States, the current training period for endovascular arterial stroke therapy requires eight or nine years after medical school, consisting of either a five year radiology residency followed by three years of fellowship; or a seven year neurosurgery residency followed by a two year fellowship. A similar training schedule is in effect in Europe. Such lengthy training periods are necessary in part because of the wide variety of techniques that must be mastered including high flow arterio-venous malformation, adhesive embolization, aneurysm coiling, large vessel sacrifice, nosebleed embolization, intracranial and carotid artery stenting, intra-arterial chemotherapy, and stroke therapy. Using simulation technology, a significant number of practice cases can be performed even in the absence of actual clinical cases, so that training time is optimally utilized and case-mix inequalities among programs can begin to be addressed.

[6] Mentice AB, Gothenburg, Sweden.

To this end, we have started the development in 2005 of a high-fidelity interventional neuroradiology simulator, aimed at training and procedure planning. The system relies on accurate patient-specific anatomical representations of the vascular anatomy and uses new algorithms for fluoroscopic rendering and physics-based modeling of catheter-vessel interactions. The full body vascular model used in our simulator consists of over 4,000 arterial and venous vessels, and is optimized for real-time collision detection and visualization of angiograms. Various aspects of the components of the system have been described in previous sections of this document, and will just be summarized here.

Technical overview

The following paragraphs briefly describe the key characteristics of the different components of the systems, as well as their integration within a prototype.

Anatomical modeling

Anatomical models of the anatomy are obtained from various modalities of contrast-enhanced medical images (MRA, CTA or 3D Rotational Angiography). While the segmentation and extraction of the skeleton from the images is partially manual, the reconstruction of the three-dimensional vascular network is automatic, given a fully connected set of central lines.

Fluid Flow

One-dimensional blood flow is computed using Poisseuille's law as in the ICTS but in the case of the cerebrovascular system, the very large number of vessels requires a complete different approach. Arterial and venous blood flow can be computed in real-time. The heart rate, as well as the systolic and diastolic pressures, are taken into account to produce a pulsatile one-dimensional flow in the arteries. Blood flow change near a stenosis is computed in real-time to illustrate the outcome of a procedure such as carotid stenting. When the contrast agent is injected, mixing between blood and contrast is computed using an advection-diffusion model and the resulting angiographic image can be rendered in real-time.

Devices

The catheter model simulates the deformation of a range of flexible devices (catheters, guidewires, coils) and their interactions with the inner arterial wall. Flexible devices are modeled as a series of beam elements. An efficient implementation of a finite element method allows real-time, accurate deformation of devices comprising more than 100 beam elements. With such a model, even highly non-linear deformations, such as the ones of coils, can be represented.

Haptics

The haptic interface device used in the ICTS was too complex to reproduce, so we opted for a commercial product from *Xitact*. This interface uses optical trackers to measure the rotation and translation of catheters and guidewires. The interface also integrates other controls typical of interventional radiology procedures, such as a foot switch, a syringe for contrast injection, and basic controls for the virtual fluoroscope.

Rendering

The method used for generating X-ray and fluoroscopic images is based on volumetric rendering of CT scan images. This presents two major advantages: first, the quality of the generated images is very close to an actual X-ray image, and second, the CT scan data is also used for generating patient-specific anatomical models of the vascular network. This ensures a perfect registration between the two, for higher fidelity simulation Also, using volume rendering techniques, the voxel data set can be deformed locally to reproduce, in the generated fluoroscopic image, cardiac or respiratory motion.

Systems integration

The complete simulation system including all of the above components was developed in C++ using a modular object-oriented approach. This system runs at interactive rates on a single CPU dual core processor (Pentium Dual Core 1.6 GHz). The different components of the system (vascular anatomy, physiology, X-ray rendering, etc.) were regularly assessed by interventional radiologists from the Massachusetts General Hospital in Boston. Through this process of rigorous development, research and discussion with physicians, we significantly improve the realism of the simulation.

***Figure 64**: Different views of a simulation of interventional neuroradiology. Left: visualization of the distribution of arterial pressures in the cerebrovascular network. Center: simulation of contrast agent injection. Right: navigation of a virtual catheter in the vascular system.*

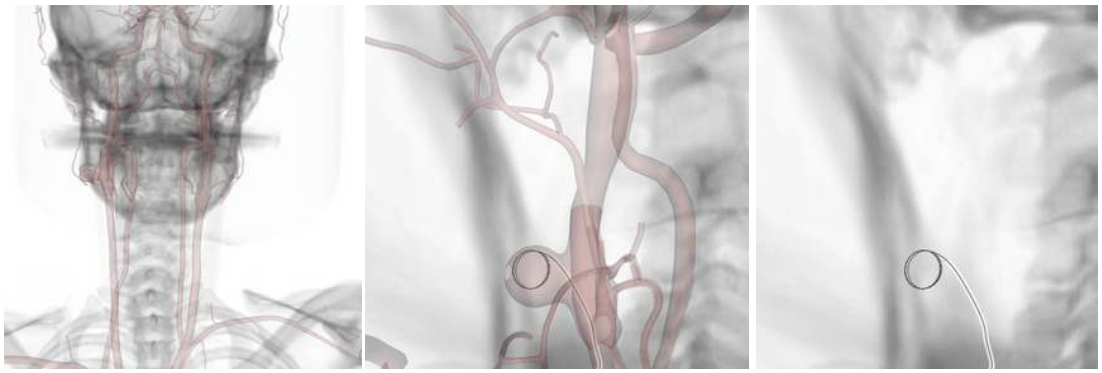


***Figure 65**: A few steps of the simulated deployment of a helical embolization coil. Left: overall view of the head with the arterial vascular systems. Center: close-up view of the coil being deployed inside a simplified aneurysm. Right: the same view, as it would be seen during a real procedure (i.e. without overlaying the vascular network to the X-ray image).*

4. Eye Surgery

Eye surgery, also known as ophthalmic surgery, involves a number of complex procedures that require a high level of training, and excellent dextrous skills. Among the more frequent and difficult procedures are: cataract surgery, glaucoma surgery, corneal surgery, vitro-retinal surgery and eye muscle surgery. The average diameter of the eye ball is 25 mm, and the lens is only about 10 mm. Incisions, and instrument motion must be very well controlled and the margin for error is very small. Although these reasons are excellent motivations for developing training systems, little work has been done in this area (Neumann *et al.*, 1998) (Barea *et al.*, 2007). One commercial product exists, the eye surgery simulator EYESi, from the company VRmagic.

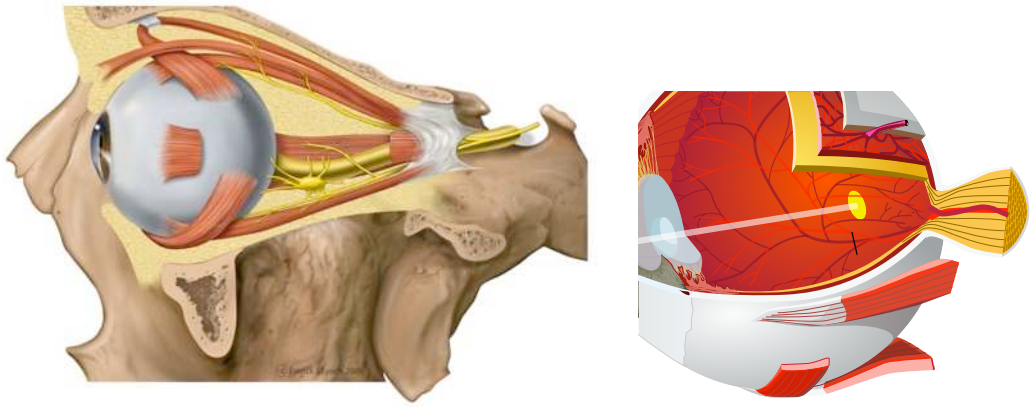


***Figure 66**: Anatomy of the eye. (Left): muscles and overall location of the eye ball in the skull. (Right): detailed anatomy of the eye, with the retina (in red), the cornea, and the lens.*

Simulation of vitrectomy

Most serious retinal eye problems that require surgery are caused by problems with the vitreous, the transparent gel-like substance which fills the center cavity of the eye (see Figures 66 and 67). The vitreous is attached to the retina, the thin layer of tissue lining the back of the eye. The retina has two parts: the peripheral retina and the macula. The macula allows to see very fine detail while the peripheral retina gives us our side vision. The vitreous is most strongly attached to the retina at the sides of the eye. It is also attached to the optic nerve, the macula and the large retinal blood vessels.

Vitrectomy is a form of eye surgery that treats disorders of the retina and vitreous. The vitreous is removed during vitrectomy and is usually replaced with balanced salt solution. A vitrectomy is required for a variety of conditions. In particular, it is done to remove scar tissue that grow on the vitreous or the surface of the retina and pulls on the retina, which causes a retinal detachment. Another pathology requiring this type of procedure is diabetic traction retinopathy, which is characterized by bleeding and scar tissue forming in the eye of a diabetic patient. Vitrectomy can also be required in case of complications following cataract surgery. In a vitrectomy, microsurgical instruments are placed into the eye through three tiny incisions in the sclera (the white part of the eye). A variety of instruments may be used to remove the vitreous gel and any scar tissue that maybe be growing on the surface of the retina. A laser probe can be inserted into the eye so that laser therapy can be performed during the surgery.

***Figure 67**: (Left): illustration of the inner anatomy of the eye. The vitreous is a transparent gel-like substance which fills the center cavity of the eye. (Right): simulation of vitrectomy surgery within the SOFA framework.*

We developed a simulation of vitrectomy surgery using the SOFA framework (see Chapter VI) where the objective of the simulation is to remove scar tissue that grows on the surface of the retina and pulls on the retina, causing a retinal detachment. This simulation uses a detailed anatomical model of the eye, a soft tissue model for the scar tissue and the retina, and a generic haptic interface (Phantom Omni). The scar tissue exerts a tensile force on the retina, thus causing a retinal detachment and consequently vision problems. The basic principle of the surgical procedure, which we replicated in the simulation, consists in making small incisions into the wall of the eye to allow the introduction of instruments into the vitreous cavity. It is followed by the incision and removal of the scar tissue using microscopic scissors, scalpels, and forceps.

Simulation of cataract surgery

A cataract is an opacity or cloudiness in the natural lens of the eye. It represents an important cause of visual impairment and, if not treated, can lead to blindness. It is actually the leading cause of blindness worldwide, and its development is related to aging, sunlight exposure, smoking, poor nutrition, eye trauma, and certain medications. The best treatment for this pathology remains surgery. Cataract surgery has made important advances over the past twenty years, and in 2005, more than 5 million people in the United States and in Europe underwent cataract surgery. Most cataract surgeries are performed using microscopic size incisions, advanced ultrasonic equipment to fragment cataracts into tiny fragments, and foldable intraocular lenses to minimize the size of the incision. All these advances benefit the patient, but increase training requirements for eye surgeons.

At the end of 2007, we started the development of a new training system for cataract surgery. The main objectives of this simulation are to reproduce with great accuracy the three main steps of cataract surgery: 1) capsulorhexis 2) phacoemulsification and 3) implantation of an intraocular lens.

- Capsulorhexis is the technique used to remove a part of the anterior lens capsule. The term describing the exact surgical technique is Continuous Circular Capsulorhexis (CCC), and uses shear and stretch forces to create an opening in the capsule.
- Phacoemulsification consists in using surgical devices which tip vibrates at an ultrasonic frequency (40,000 Hz) resulting in a local emulsification of the lens material. A second fine

instrument may be used from a side port to facilitate cracking the lens nucleus (usually stiffer than the peripheral part of the lens) into smaller pieces, to make the emulsification easier. After phacoemulsification of the lens nucleus and cortical material is completed, a dual irrigation-aspiration is used to aspirate out the remaining peripheral cortical material.

- After the removal of the diseased lens, an intraocular lens is implanted into the eye, through a small incision (about 2 mm) using a foldable intraocular lens. The foldable lens, made of silicone or acrylic material, is folded using a proprietary insertion device provided along with the intraocular lens. It is then implanted within the lens capsule through the same incision as used during phacoemulsification.

Although a first proof of concept had been developed at INRIA by the Alcove team in 2005 and 2006, several limitations in the technical choices required a new development. We chose to develop this new simulation using the SOFA framework, both to have a more flexible choice of technical solutions, and also to test the prototyping capabilities in SOFA. The anatomical model of the eye, created by P. Neumann from CIMIT, was quickly integrated (Figure 68) and the main focus went to the choice of appropriate deformable models for the lens and lens capsule. An important effort also went into the development of topological changes corresponding to the capsulorhexis and phacoemulsification.

***Figure 68**: anatomical model of the eye in the cataract surgery simulation. (Left): the anatomical model in SOFA is rather detailed, and (Right) an illustration of a phacoemulsification device interacting with the tetrahedral mesh of the lens. The particular rendering of the lens is only for the purpose of showing the size of the tetrahedral elements.*

The new version of the simulator integrates a non-linear tetrahedral Finite Element Model for the lens, and a non-linear triangular Finite Element Model for the capsule. Both models are based on a co-rotational method, and are freely available in SOFA. To add resistance to bending to the capsule, rotational springs are added to the FEM model. Instrument models were then created, in particular a micro-scalpel, micro-graspers and a phacoemulsification device. Each device can trigger a particular action when applied to a particular anatomical structure. Currently, it is possible to make an incision in the capsule with a micro-scalpel, and then tear the capsule by pulling on the capsule with a grasper, near the incision. The current model does not account for the circular arrangement of fibers in the capsule that allow to perform a circular capsulorhexis. Currently, tissue fracture is only based on principal directions of the strain tensor. Finally, the phacoemulsification is simulated by removing tetrahedral elements in the FEM mesh when they are in contact with the virtual phacoemulsification device. The resulting visualization is currently too

crude, as the tetrahedral mesh is rather coarse, with 814 nodes and 4059 tetrahedra (see Figure 68). Instrument tracking, as well as basic haptic feedback, is possible using a PHANToM device (Omni). For more realistic interactions, this low-cost, generic system should be replaced by a dedicated tracking interface, with very high tracking resolution.

The next stage in the development of the simulation system will consist in integrating a GPU-based version of the FEM algorithm within the simulator. The GPU-based version of the co-rotational FEM model currently allows for a speedup factor of about 15. This would allow using a more detailed mesh for the lens (Figure 69), leading to a more realistic phacoemulsification simulation. Additional work will relate to the modeling of the Continuous Circular Capsulorhexis, as well as more realistic rendering of the eye and surgical instruments.

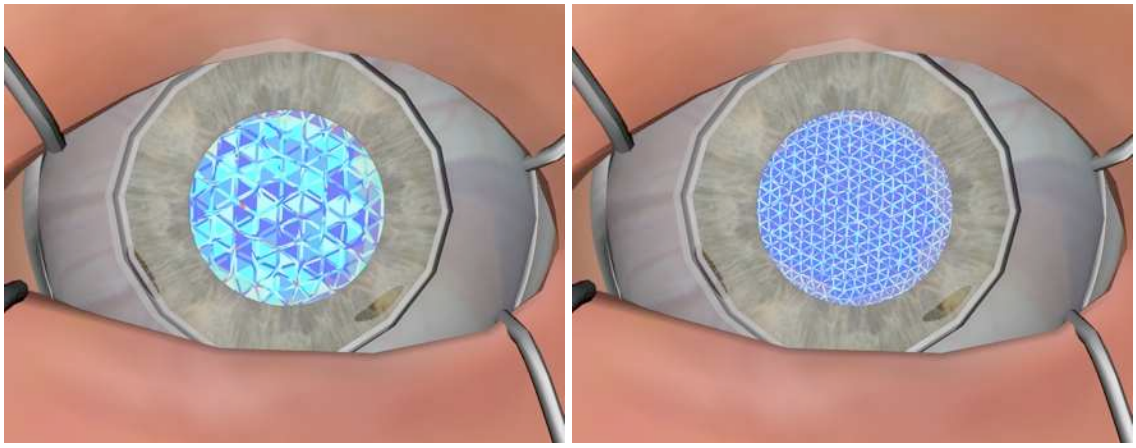


***Figure 69**: (Left): illustration of the current resolution of the FEM mesh of the eye lens. (Right): using the GPU, a speedup factor of about 15 can be obtained for the FEM algorithm. Consequently, a higher resolution mesh could be used without increasing computation times, and would result in a more realistic simulation of phacoemulsification.*

Summary of contributions

- **Patents**
  - **Cotin** S, Delingette H, Ayache N; "Electronic Device for processing image-data for simulating the behavior of a deformable object", International patent WO9926119A1, 1998
  - Dawson S, **Cotin** S, Ottensmeyer M, Neumann P; "Medical Training System", United States Patent Application, 10/488,415 - United States and European patent application PCT/US02/28593, 2004.
  - **Cotin** S, Stylopoulos N, Dawson S, Ottensmeyer M; "Surgical Training System for Laparoscopic Procedures", United States Patent Application, 10/797874, 2004.
  - **Cotin** S, Bardsley R, Dawson S ; "Design of a new haptic/tracking interface for interventional radiology training", Massachusetts General Hospital Invention Disclosure
  - **Cotin** S, Wu X, Neumann P, Dawson S ; "Methods and Apparatus for Simulation of Endovascular and Endoluminal Procedures", United States Patent Application, 60/600,188 - PCT application number PCT/US2005/028594, 2005.
- **Software**
  - **Interventional Cardiology Training System:** development time : 15 months; more than 20,000 lines of code; co-developers: 3 persons. This code served as the core for the development of a commercial product named VIST (www.mentice.com).
  - **Chest Trauma Training System:** simulator combining a real-size patient mannequin, electromagnetic position sensors, and a computer system. VIRGIL won the prestigious Army's Greatest Inventions Award in 2004. Development time: 12 months; 2 developers; about 10,000 lines of code.
  - **Computer Enhanced Laparoscopic Training System:** implementation of a new concept for performance metrics and scoring based on kinematics analysis; development of a fully functional prototype that passed several validation tests, and was ranked first in a validation study involving several existing commercial products. Development time: 18 months; unique developer; about 5,000 lines of code.
  - **Training System for Interventional Radiology:** this project, which goal is to develop a very high-fidelity simulation system, is currently targeted at stroke therapy. However, the models and solutions introduced in this prototype can be applied to any interventional radiology specialty, from cardiology to abdominal interventions. Development time: 24 months; about 30,000 lines of code; co-developers: 5 persons.
- **Awards**
  - The "Edward M. Kennedy Award" for Health Care Innovation, Partners Health Care, Boston
  - Top Ten Army's Greatest Inventions Award, US Department of Defense, Washington DC
  - Several prototypes developed at CIMIT were chosen to become part of the permanent collection at the National Museum of Health and Medicine in Washington, DC.
- **Miscellaneous**
  - Numerous demonstrations at scientific, medical or general meetings of the different training systems described in this chapter.
  - Several advanced discussions about licensing and commercializing the different training systems described in this chapter.

Chapter VI
A Unified Framework for Medical Simulation

1. Introduction

SOFA, the Simulation Open Framework Architecture, is an international, multi-institution, collaborative initiative, aimed at developing a flexible and open source framework for Medical Simulation. This will eventually establish new grounds for a widely usable standard system for long-term research and product prototyping, ultimately shared by many academic and industrial sites. The development of SOFA as a concerted project started in October 2004, based on ideas introduced in an earlier project entitled CAML which I initiated while I was with the Sim Group at CIMIT. Over the last two years, the SOFA framework has evolved from an informal collaborative work between the Sim Group at CIMIT, the Alcove, Asclepios and Evasion teams at INRIA[7] into a more structured development project. By proposing a unique architecture allowing the integration of the multiple competencies required for the development of a medical training system, we believe it will be possible to accelerate and foster research activities in the field of interactive medical simulation. Currently, the development of SOFA is in direct link with research activities conducted at INRIA, at CIMIT in Boston, at ETH Zurich, at CSIRO in Australia, at the Imperial College and UCL in England, and in several other research groups. The different research activities conducted by these groups are essential to the rapid growth of the field on computer-based medical simulation.

The main objective of the SOFA project is to facilitate the development of novel and computationally efficient algorithms, and to simplify their transfer toward the clinical and industrial worlds. As a consequence, we believe SOFA could also foster collaborations among research groups by proposing a core technology that will support the integration of new components in an efficient manner. In summary, the objectives of the SOFA framework are to:

- Simplify the development of medical simulation systems by improving interoperability
- Promote collaboration between research groups
- Evaluate and validate new algorithms
- Accelerate the prototyping of simulation systems by promoting component reusability
- Facilitate technology transfer between research and industry

To achieve these objectives it is mandatory to initiate and strengthen research activities in the field of Medical Simulation. Through the joint efforts of CIMIT and INRIA, we have developed an efficient framework and a large number of modules that already provide numerous functionalities in the field of interactive simulation. We believe we can leverage this initial effort to develop new research-oriented collaborations across multiple academic groups, which would eventually lead to the creation of a consortium of academic, industrial, and clinical partners. To emphasize this strategy, we have proposed

[7] Research groups from ETH Zurich, UCSF, and Case Western Reserve University have also participated to specific developments or discussions related to SOFA.

the creation of a national research initiative, with the support of INRIA. The principal aim of this national initiative is to use the same collaborative approach we have used for the development of SOFA, but in this case applied to research problems related to medical simulation. Although the primary objective of the national initiative is to strengthen research activities in the field of medical simulation, the existence of the SOFA framework will simplify the integration of the newly developed algorithms while offering a mean to compare and validate results. Combining our research and development efforts, we propose an agenda geared towards three main objectives (as illustrated by Figure 70):

- Research and development of new algorithms dedicated to Medical Simulation
- Continuing development of the SOFA framework
- Development of a series of medical simulators as proofs of concept of our research activities

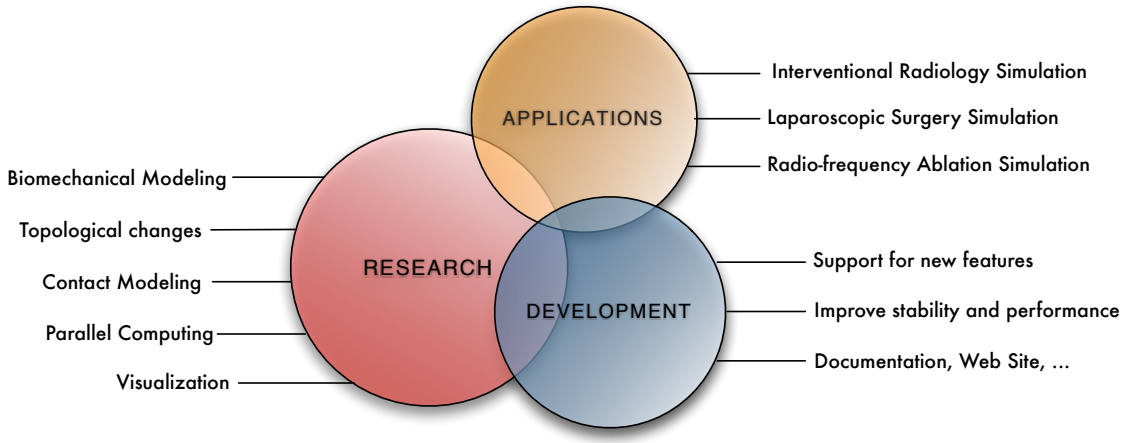


***Figure 70**: Multidisciplinary research and development of the SOFA framework need to take place simultaneously to quickly advance research in the field of computer-based interactive medical simulation. program. A series of applications will be created to illustrate and validate our research, and show the potential of the SOFA framework.*

2. The SOFA framework

For the past few years, there have been a few attempts at designing software toolkits for medical simulation. Examples include SPRING (Montgomery *et al.*, 2002), GiPSi (Goktekin *et al.*, 2004), SPORE (Meseure *et al.*, 2003), VRASS (Kawasaki et al., 2005), or SSTML (Bacon *et al.*, 2006). These different solutions aim at the same goal: providing an answer (usually Open Source) to the various challenges of medical simulation research and development. Although our aim is identical, we propose a different approach, through a very modular and flexible software framework, while minimizing the impact of this flexibility on the computation overhead. To achieve these objectives, we have developed a new architecture that implements a series of innovative concepts. Also, by developing the SOFA framework collaboratively with scientific experts in the different areas of medical simulation, we believe we can provide state-of-the-art solutions that are generically applicable, yet computationally efficient. The following sections describe in more details our approach to the development of this framework, from a technical standpoint and from the perspective of a collaborative work.

SOFA :: Architecture

As we have seen in previous chapters, medical simulation relies on a variety of interacting physics-based models, such as rigid structures (e.g. bones), deformable structures (e.g. soft-tissues) and fluids. It also involves anatomical representations through geometrical models, used for visual rendering, collision detection or for generating meshes that will support various computational models. Finally, interactions between these different models need to be efficient, accurate and capable of handling a variety of representations. In some instances, a hierarchy also exists between the various anatomical structures, and needs to be taken into account in the description of the simulated environment. The design of the SOFA architecture, by supporting these various requirements, brings the flexibility needed for academic research. Yet, its very efficient implementation makes it also suitable for professional applications and potentially for product development.

The SOFA architecture relies on several innovative concepts, in particular the notion of multi-model representation. In SOFA, most simulation components (deformable models, collision models, medical devices, etc.) can have several representations, connected through a mechanism called mapping (see Figure 71). Each representation is optimized for a particular task (e.g. collision detection, visualization) while at the same time improving interoperability by creating a clear separation between the functional aspects of the simulation components. As a consequence, it is possible to have models of very different nature interact together, for instance rigid bodies, deformable objects, and fluids. This is an essential aspect of SOFA, as it will help the integration of new research components. This modular design also facilitates the rapid prototyping of simulation systems, allowing various combinations of algorithms to be tested and compared against each other.

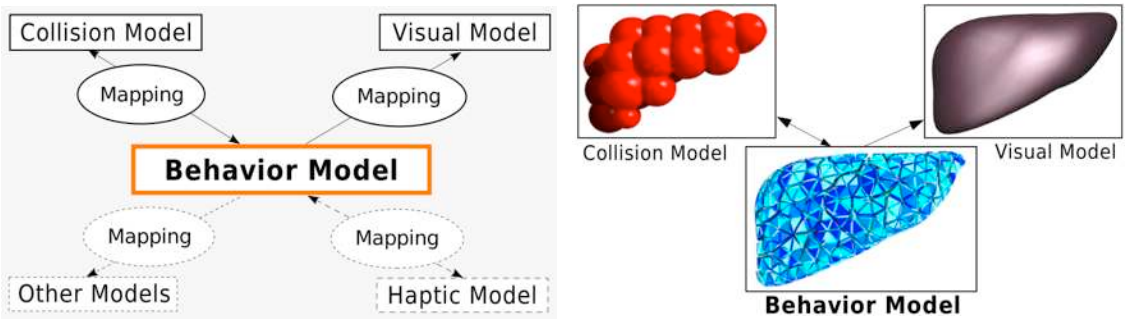


***Figure 71**: Illustration of the multi-model representation in SOFA. Left: possible representations for a simulated object, with the Behavior Model controlling the update of the other representations through a series of mappings. Right: examples of these representations for a liver model. Notice how the Visual Model is more detailed than the Behavior Model and how the Collision Model relies on a very different representation.*

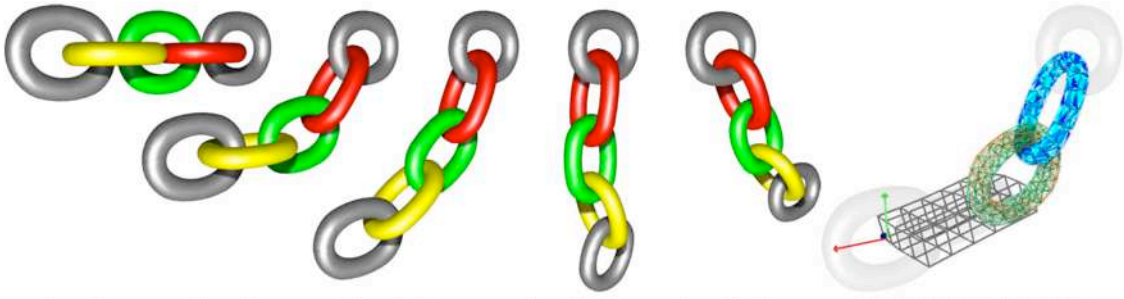
At a finer level of granularity, we also propose a decomposition of physical models (i.e. any model that behaves according to the laws of physics) into a set of basic components. In the case of (bio)mechanical models, which are computationally expensive, many strategies have been used to improve computation times or to reduce the complexity of the original model: linear elastic models have often been used instead of more complex non-linear representations, mass-spring methods as an alternative to finite element methods, etc. Each of these simplifications induces drawbacks, yet the importance of these drawbacks depends largely on the context in which they are applied. It becomes then very difficult to choose which particular method is most likely to provide the best results for a given simulation. To

address this issue in SOFA we have introduced, for the Behavior Model, a finer level of granularity than what is described in Figure 71. This permits for instance to try different time integration schemes to see the change in performance or robustness of the simulation, or to test different constitutive models. These changes can be made in a matter of seconds, without having to recompile any of the code, by simply editing an XML file. The example below allows the simulation of a deformable model of the liver, using a co-rotational FEM model as a Behavior Model, a set of spheres as Collision Model and a triangulated mesh as a Visual Model. The Visual Model is connected to the Behavior Model using a barycentric mapping.

```
<Node name="root" dt="0.02" showCollisionModels="0" showForceFields="0">
   <Object type="Simulation" name="Simulation"/>
   <Object type="CollisionPipeline" verbose="0"/>
   <Object type="BruteForceDetection" name="N2"/>
   <Object type="CollisionResponse" response="default"/>
   <Node name="Liver">
         <Object type="CGImplicit" iterations="25"/>
         <Object type="MechanicalObject" template="Vec3f" name="Liver"
                 filename="BehaviorModels/liver.xs3"/>
         <Object type="DiagonalMass" name="mass" filename="BehaviorModels/liver.xs3"/>
         <Object type="Mesh" name="meshTopology" filename="Topology/liver.mesh"/>
         <Object type="TetrahedronFEMForceField" name="FEM" youngModulus="500"
poissonRatio="0.3" computeGlobalMatrix="false" method="large"/>
         <Object type="FixedConstraint" name="FixedConstraint" indices="3 39 64"/>
         <Node name="Visu">
               <Object type="OglModel" name="VisualModel" filename="VisualModels/liver-
smooth.obj" color="red"/>
               <Object type="BarycentricMapping" object1="../.." object2="VisualModel"/>
         </Node>
         <Node name="Surf">
               <Object type="Sphere" name="CollisionModel" filename="CollisionModels/
liver.sph"/>
               <Object type="BarycentricMapping"/>
         </Node>
   </Node>
</Node>
```

To achieve this level of flexibility, we have defined a series of generic primitives, or components, that are common to most physics-based simulations: DoF, Mass, Force Field, and Integration Scheme and Linear Solver. The DoF component describes the degrees of freedom (and their derivatives) of the object. This includes positions, velocities, accelerations, as well as other auxiliary vectors. The Mass component represents the mass of the object. Depending on the model, the mass can be represented by a single value (all the DoFs have the same mass), a vector (the DoFs have a different mass), or even a matrix as used in complex finite element models. The Force Field describes both internal forces associated with the constitutive equations of the model, and external forces that can be applied to this object. A variety of forces are currently derived from the abstract Force Field representation, including springs, linear and co-rotational FEM, Mass-Tensor, and Smoothed Particle Hydrodynamics (SPH). The Integration Scheme component handles time integration, i.e. advancing the state of the system from time t to time $t+\Delta t$. To this end, requests are sent to the other components to execute operations such as summation of forces, computation of accelerations, and vector operations on the DoFs such as $x=x+v\cdot\Delta t$. The Linear solver then solves the resulting system of equations. Currently SOFA integrates many different Force Field components, time integration schemes, and solvers.

Building and maintaining the relations between all the elements of a simulation can become quite complex. Reusing concepts from the graphics community, we decided for a homogeneous scene-graph representation, where each component is attached to a node of a tree structure. While components are user-defined and can be extended at will, internal nodes are all the same. They only store pointers to their local components, as well as their parent and children nodes. This simple structure enables to easily visit all or a subset of the components in a scene, and dependencies between components are handled by retrieving sibling components attached to the same node. For instance, a Force Field component can access the DoF component by getting its pointer from the node. The scene-graph can also be dynamically reorganized, allowing for instance the creation of groups of interacting objects. Such groups can then be processed as a unique system of equations by the solver, thus permitting to efficiently handle stiff contact forces. Another advantage of using a scene-graph is that most computations performed in the simulation loop can be expressed as a traversal of the scene-graph. This traversal is called an action in SOFA. For instance, at each time step, the simulation state is updated by sending actions to perform time integration and then solve the resulting system. Each solver then forwards requests to the appropriate components by recursively sending actions within its sub-tree.

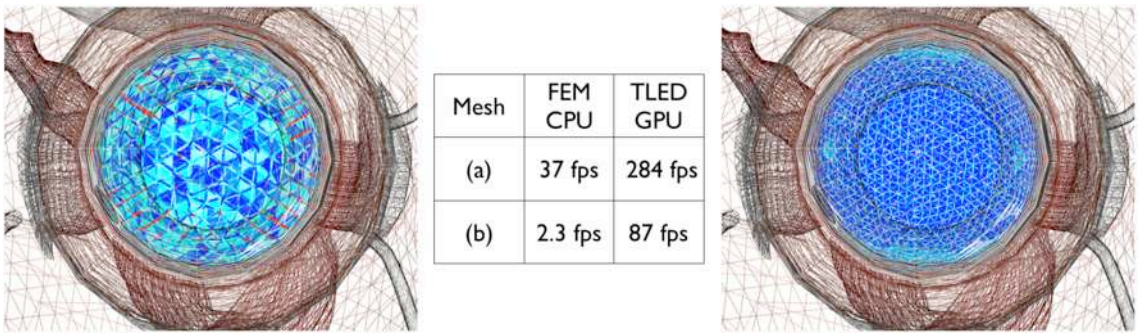


t=0 *t=1.0s* *t=1.5s* *t=2.0s* *t=2.5s*

***Figure 72**: Animation of a chain combining a FEM model, a mass-spring model, a FFD grid, and a rigid body. This example is a perfect illustration of the flexibility of SOFA. Not only several algorithms for rigid or deformable bodies can be part of the same simulation, but they can also interact in a physically correct manner. No constraints between links were pre-defined, instead we relied on collision detection and stiff contact forces to handle the contacts. Using implicit integrator handling dynamically-created groups of interacting objects resulted in a stable simulation.*

GPU support

GPUs are parallel processors dedicated to three-dimensional graphics, some of them comprising nearly two hundred computing units. These architectures have recently been exploited in different scientific domains requiring high computational performances. GPUs can be available in regular workstations, and in cluster setups. As a step toward the future integration of more complex biomechanical models in SOFA, we have added a support for the CUDA library from NVIDIA which provides a simplified approach to GPU programming. Several preliminary results of GPU-based simulations have been obtained, permitting to reach speedup factors (compared to a single core GPU) ranging from 16 to 55 depending of the model, integration scheme and other factors (see Figure 73). Such improvements permit to also consider simulations with finer meshes for the simulated objects, new algorithms for contact modeling or for simulating physiological processes. However, while the fast evolutions of parallel architectures is useful to increase the realism of simulations, their variety (multi-core CPUs, GPUs, clusters, grids) make the design of parallel algorithm challenging. An important effort in the development of SOFA consists in minimizing the dependency between simulation algorithms and hardware architectures.



Mesh	FEM CPU	TLED GPU
(a)	37 fps	284 fps
(b)	2.3 fps	87 fps

(a) 814 nodes, 4058 elements (b) 7826 nodes, 47633 elements

***Figure 73**: Comparison of cataract surgery performance using co-rotational FEM on CPU and TLED on GPU. Computation times are measured in SOFA. The TLED algorithm is one of the several algorithms implemented on the GPU and freely available with SOFA.*

SOFA :: Development

As mentioned previously, an essential characteristic of SOFA is its collaborative design and development. Nonetheless, to remain efficient, the development process needs to be structured. On a day-to-day basis, the management of the development activities relies on several collaborative development tools. SOFA is currently hosted on the INRIA gForge, with controlled access to the development branch of the SOFA code. There are about 10 active developers and another 10 occasional developers. These developers are mostly INRIA researchers or software engineers. Version control for the source code is based on SVN, a popular version control system. Recently we have started experimenting the use of an IRC client to facilitate communication between developers. For this purpose we use a dedicated channel for SOFA (provided by www.freenode.net). Also, since the early stages of the development of SOFA we have organized weekly audio-conferences to discuss progress, ideas, problems and set new tasks. Reporting for the weekly discussions is available on the private Wiki page of the SOFA web site. These audio-conferences are combined with meetings every three to four months. These meetings are usually hosted in Lille, Grenoble or Sophia-Antipolis, and involve 15 to 20 people, permanent researchers, Ph.D. students, postdoctoral students, and developers.

Automated Continuous Integration

When developing on different platforms with different operating systems, it becomes quickly cumbersome to frequently test if the newly committed code compiles under each of these architectures. To support an increasing number of platforms and operating systems, we have developed a compilation dash board. It consists in a dedicated machine that uses virtual machines to run Windows, Linux and Mac OS X operating systems. That machine regularly checks the SVN server to see if new code has been committed. When it is the case, the entire SOFA code is automatically compiled, and errors or warnings are reported on a web page. Scheduled builds are performed frequently enough to qualify as continuous integration. The build script not only compiles binaries but also generates documentation, and website pages. Continuous integration has many advantages, although compilation times for the whole SOFA code, on several architectures, tends to become time consuming. This requires the development of compilation strategies to accelerate build times. Currently, everything is rebuilt at each new commit to ensure that someone downloading SOFA will be able to build everything without problem.

Web Site

The SOFA website (www.sofa-framework.org) is an important element in the distribution of the framework to the simulation community. With research articles, it constitutes an important way to communicate our progress and results to the scientific community. But the website also serves as a portal where documentation, examples, tutorial and other information are regularly updated and that current users of SOFA can easily access. Finally, the SOFA website will allow a community of users to be developed, and projects based on SOFA to be highlighted.

SOFA license

The choice of the license has been discussed with MGH, which is a partner in the development of SOFA. The architecture of SOFA is composed of a core and a set of modules. The core of SOFA is protected under a LGPL license, as well as the currently available modules. However, we allow our partners to develop independent modules through a different licensing mode as long as they share the modifications and improvements made on the core of SOFA.

Current status

Version 1.0 beta 1 of SOFA was released in February 2007 during the MMVR conference in Long Beach, California. It was soon followed by version 1.0 beta 2. Nearly 25,000 downloads of the beta 2 have been counted as of May 1st, 2008. Currently, thanks to its advanced architecture, SOFA allows to:

- Create complex and evolving simulations by combining new algorithms with existing algorithms
- Modify most parameters of the simulation by simply editing a XML file
- Build complex models from simpler ones using a scene-graph description
- Efficiently simulate the dynamics of interacting objects using abstract equation solvers
- Reuse and easily compare a variety of available methods
- Transparently parallelize complex computations using semantics based on data dependencies
- Use new generations of GPUs through the CUDA API to greatly improve computation times

Several results obtained using SOFA can be seen on the SOFA website at www.sofa-framework.org/gallery. Most of these results are generic and only aim at validating the different aspects of the SOFA framework. Developments of complex medical simulations have recently started, in particular in the areas of ophthalmic surgery and interventional radiology. In addition, a few simple examples of medical simulations have been developed as testbeds for new algorithms. The current public version of SOFA includes the following modules, contributed at various degrees by a team of 20 people, for a total of nearly 200,000 lines of code:

- **Deformable models**: mass-springs, linear and co-rotationnal FEM
- **Rigid models**: articulated bodies based on penalty method or reduced coordinates
- **Fluid models**: Smoothed Particles Hydrodynamics, Eulerian approaches
- **Collision models**: spheres, triangular meshes, distance fields (preliminary); with AABB-tree or octree bounding volume hierarchies
- **Collision detection methods**: proximity, continuous (preliminary)

- **Collision response methods**: (implicit) penalty method, LCP-based constraints
- **Integration schemes**: explicit Euler, 2nd and 4th order Runge-Kutta, implicit Euler using a Projected Conjugate Gradient
- **Linear solvers**: conjugate gradient, Gauss-Siedel, band matrix solver, LU decomposition

Principal Milestones

To become an internationally acknowledged research tool for medical simulation, important milestones need to be met:

- **Milestone #1 - end of 2009**: develop a series of prototypes of simulators (Interventional Radiology, Laparoscopic Surgery, Eye Surgery, ...) based on code developed by the different teams involved in the SOFA project. This will test the flexibility of SOFA to support various algorithms in a unified environment. If successful, it will show that it is possible to develop, using SOFA, simulations of the same quality than what is currently possible using dedicated solutions. The effort required to reach this stage is estimated to take about two years, with four full-time software engineers and limited contributions from the researchers of the various teams. At this point, the core of SOFA and its documentation, should be stable enough to facilitate the integration of third-party algorithm within SOFA as well as the development of various applications.
- **Milestone #2 - end of 2011**: this second milestone is tightly related to the progress on the research side of the SOFA project. The main objective is the integration of new algorithms developed by researchers from INRIA and other institutions collaborating on the SOFA project, to improve the realism of three simulation systems: training and planning for laparoscopic hepatectomy; training and planning system for interventional neuroradiology; training and planning system for cardiac radio-frequency ablation.

While at the end of Milestone #1 the simulations developed with SOFA should be on par with the state-of-the-art in Medical Simulation, the ultimate objective at the end of Milestone #2 is to create the most advanced simulations in the field. If this goal is met, SOFA will be regarded as a tool of high value by both the research community, the industry, and the clinical community.

Some results

The development of SOFA started at the end of 2005. In February 2007 we demonstrated our first results and released the first public version. As of May 2008, we have developed a large set of functionalities, and have used SOFA as a central piece in several publications to international conferences (SIGGRAPH, MICCAI, ISBMS, and CGI). Using SOFA helped validate our work and develop demonstrations used in videos supporting our submissions without much additional work since it was possible to reuse many existing components. More importantly, using SOFA only adds an average of 5% overhead which means benchmarks done using SOFA remain relevant.

Below are several examples of simulations (not all of them in the field of medical simulation) recently developed using SOFA.

***Figure 74**: Simulation of ophthalmic surgery in SOFA, demonstrated at MMVR 2007. The simulated procedure is called vitrectomy.*

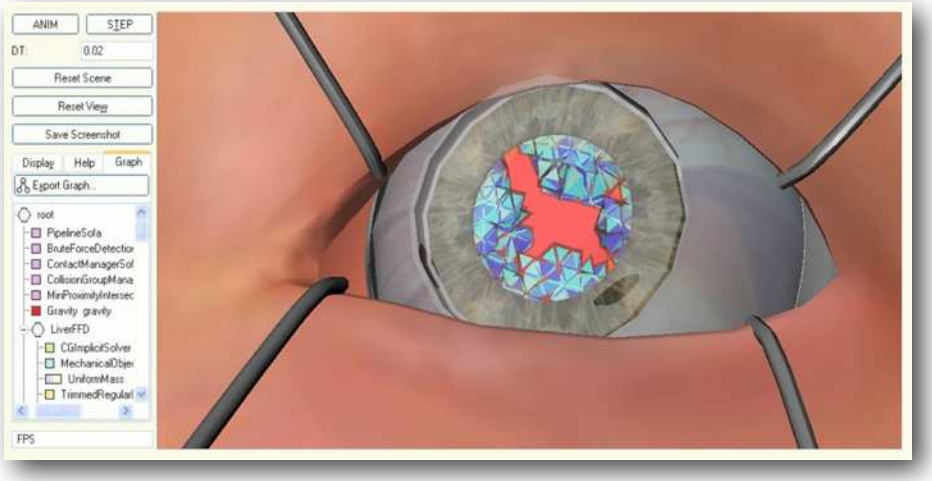


***Figure 75**: Simulation of cataract surgery in SOFA. Only 5 weeks were necessary to develop this prototype with SOFA, while the previous one required almost a year.*

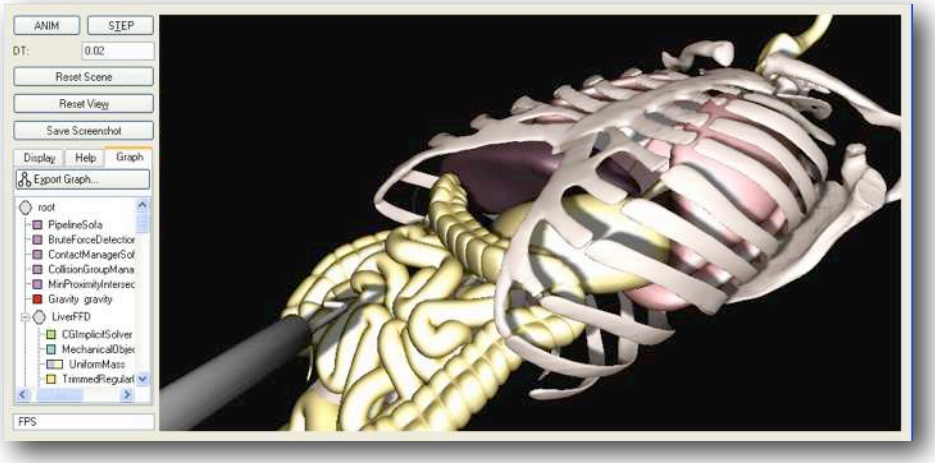


***Figure 76**: Simulation of laparoscopic surgery in SOFA, with haptic feedback. Demonstrated at MMVR 2007 during a workshop on open source toolkits.*

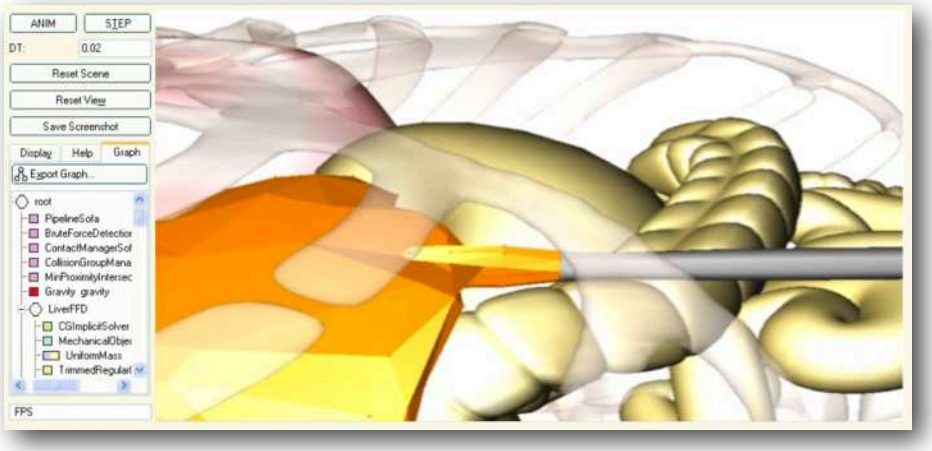


***Figure 76**: Realistic grasping of soft-tissue using an accurate contact model with friction. If the instrument is pulled back with enough force to overcome the friction, it will release the organ.*

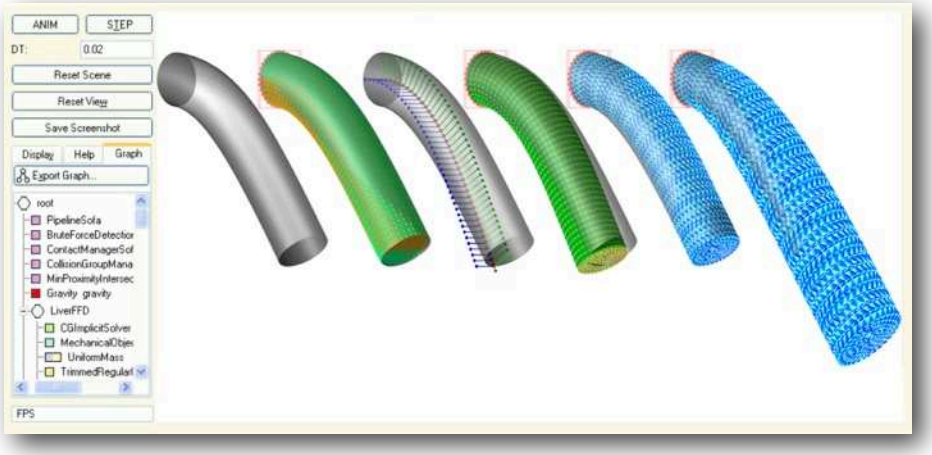


***Figure 77**: Cross-comparison of deformable models in SOFA. Models can be compared to each others and also quantitatively validated against a reference model.*

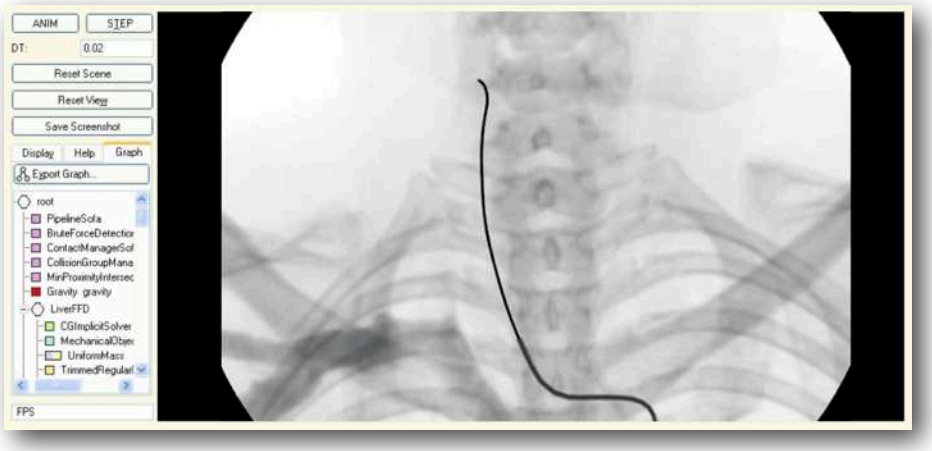


***Figure 78**: Simulation of interventional radiology in SOFA using a dynamic non-linear FEM model of catheter (guidewires and coils can also be simulated with the same approach).*

3. SOFA team

The SOFA team is currently composed of scientists and software engineers from INRIA, CIMIT, and CSIRO that collaboratively develop the SOFA framework, and communicate regularly through weekly audio-conferences as well as quarterly meetings. Other groups join these meetings and collaborate on a occasional basis. The SOFA team is managed by three people, the project leader (myself), the chief architect (Francois Faure) and the chief developer (Jeremie Allard). This group of people is globally responsible for the reporting aspects, financial management, supervision of development activities or management of important milestones. On the other hand, the validation of new architecture designs, the choice of new functionalities, and many other development aspects are done in a collegiate manner.

The ALCOVE, EVASION and ASCLEPIOS teams have been involved, with the Sim Group in Boston, in the design and development of the SOFA framework since 2005. These teams have a long history of collaboration, and have efficiently contributed to the current version of the SOFA framework. Recently the MOAIS team has joined the development effort on the parallelism aspects. To increase this initial effort and make SOFA a complete and efficient research tool for Medical Simulation, we need to 1) improve the design of the architecture, 2) develop new functionalities, 3) create a series of exemplary applications in the field on Medical Simulation, and 4) reinforce and structure development activities. The five leading teams on this effort are listed below:

- **ALCOVE** (https://www.inria.fr/recherche/equipes/alcove.en.html)
- **ASCLEPIOS** (https://www.inria.fr/recherche/equipes/asclepios.en.html)
- **EVASION** (https://www.inria.fr/recherche/equipes/evasion.en.html)
- **CIMIT Sim Group** (http://www.medicalsim.org/)
- **MOAIS** (https://www.inria.fr/recherche/equipes/moais.en.html)

To evolve as a reference tool for the Medical Simulation community (and potentially other communities interested in real-time physics-based simulation), SOFA needs to integrate state-of-the-art algorithms. While the role of the design team is to develop, maintain and improve the core of SOFA, it is important that SOFA functionalities evolves constantly to remain close to the state-of-the-art in the field. This cannot be achieved only by the teams listed above but requires an international, collaborative effort. We have already developed collaborations with several groups that intend to bring new algorithms into SOFA. Below is the list of teams that have agreed to contribute to this overall effort or have already contributed:

- **SIM Group @ CIMIT**: the SIM Group was formed in 1998 as a core focus area of the Massachusetts General Hospital and CIMIT, a technology development and incubation laboratory located in Boston. Sim Group members have a multidisciplinary expertise in medicine, mechanical engineering, biomedical engineering, electrical engineering, computer science, and industrial design. Research done in the group covers various aspects ranging from soft-tissue characterization to medical device modeling, metrics for surgical skills, to interventional radiology simulation. Current plans for integration of new algorithms within SOFA include: performance metrics and adaptive learning tools.

- **Biomedical Imaging group, CSIRO**: this group is developing new tools to analyze and extract valuable information from medical images, such as magnetic resonance imaging (anatomical, functional, spectroscopy), computed tomography, positron emission tomography, ultrasound imaging, molecular imaging, and histology imaging. The main application domains are: image-guided therapy, image-guided surgery, surgery simulation, computer aided diagnosis, therapy monitoring. An important application being currently developed is a colonoscopy training system using nonlinear elastic models computed on the GPU. The main objective in the short term is to 1) implement using the CUDA API a GPU-based nonlinear elastic model, and 2) add this new algorithm to the list of modules in SOFA. In the longer term, other algorithms (resulting from co-supervised Ph.D work) will be included in SOFA.
- **Virtual Reality in Medicine Group, ETH**: this group has an important expertise in the area of medical simulation, in particular through the development of a hysteroscopy simulator. This group has been involved very early in discussion about SOFA and has contributed FEM code that was integrated in an early version of SOFA.
- **Laboratoire TIMC-IMAG**: the Computer-Aided Surgery group (GMCAO: Gestes Médico-Chirurgicaux Assistés par Ordinateur) of TIMC laboratory aims at assisting surgeons both during the diagnostic and therapeutic aspects of a procedure, in order to improve safety and accuracy of the intervention. In this framework, numerical models of human anatomical structures have been developed to help surgeons plan the procedure, and eventually provide per-operative guidance. Some of the work done in this group, under the supervision of Yohan Payan, has been partially integrated within SOFA. It includes a hierarchical multi-resolution finite element model, as well as a co-rotational finite element model.

Towards a consortium

The main objective of the development of the SOFA framework is the creation of a strongly connected community of researchers in the field of Medical Simulation. Through the efforts of the past two years we have created an efficient platform where many algorithms are already available in most areas of interactive simulation (collision detection, rendering, deformable models, contact modeling, topological changes, ...). Recently, other groups have started contributing to SOFA (CSIRO biomedical imaging group, CEA interactive simulation group, LSIIT Computational Geometry & Computer Graphics, and the MAGRIT team at INRIA). During the past year, several established or startup companies also indicated an interest in using SOFA, for applications in medical simulation, engineering, and gaming. Finally, several hospitals or clinical institutions are directly or indirectly involved in the SOFA project, through collaborations with members of the SOFA team. These are world-renowned institutions in areas such as laparoscopic surgery, interventional radiology, or radio-frequency ablation, but they also have a strong interest in medical simulation. The current list includes: IRCAD in Strasbourg, MGH in Boston, Centre Hospitalier Universitaire of Nancy, and Guy's Hospital in London.

Summary of contributions

- **Research articles**
  - About SOFA
    - J. Allard, S. Cotin, F. Faure, P.-J. Bensoussan, F. Poyer, C. Duriez, H. Delingette, and L. Grisoni. “SOFA – an Open Source Framework for Medical Simulation”. In Medicine Meets Virtual Reality (MMVR), 2007.
    - F. Faure, J. Allard, S. Cotin, P. Neumann, P.-J. Bensoussan, C. Duriez, H. Delingette and L. Grisoni. “SOFA - a Modular Yet Efficient Simulation Framework”. Proc. Surgetica, 2007.
  - Made possible using SOFA
    - J. Dequidt, M. Marchal, C. Duriez, and S. Cotin. “Interactive Simulation of Embolization Coils: Modeling and Experimental Validation”. Submitted to the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), 2008.
    - G. Saupin, C. Duriez, and S. Cotin. “Contact model for haptic medical simulations”. In Proc. International Symposium on Computational Models for Biomedical Simulation - ISBMS, 2008 (to appear).
    - M. Marchal, J. Allard, C. Duriez, and S. Cotin. “Towards a Framework for Assessing Deformable Models in Medical Simulation”. In Proc. International Symposium on Computational Models for Biomedical Simulation - ISBMS, 2008 (to appear).
    - O. Comas, Z. Taylor , J. Allard, S. Ourselin, S. Cotin, and J. Passenger. “Efficient nonlinear FEM for soft tissue modeling and its GPU implementation within the open source framework SOFA”. In Proc. International Symposium on Computational Models for Biomedical Simulation - ISBMS, 2008 (to appear).
    - G. Saupin, C. Duriez, L. Grisoni, S. Cotin. “Efficient Contact Modeling using Compliance Warping”. In Proc. Computer Graphics International, 2008 (to appear).
- **Software**
  - **SOFA - Simulation Open Framework Architecture**: this project is an international, multi-institution collaborative initiative, aimed at developing an open source and extendible framework for Medical Simulation. The main objective of SOFA is to foster collaboration among research groups and facilitate the development of new prototypes. The SOFA project is still under developments. About 200,000 lines of code have been written collaboratively by 15 scientists and software engineers. The core and initial framework was co-developed by a small team of researchers from INRIA and CIMIT. The SOFA project is, at the beginning of 2008, the 2nd most downloaded software on the INRIA gForge, with more than 20,000 downloads.
- **Miscellaneous**
  - Collaboration with CSIRO to integrate a very efficient GPU-based FEM algorithm into SOFA.
  - European project PASSPORT for Liver Surgery (awarded in 2008) where SOFA will serve as the backbone for the integration of all simulation components.

Chapter VII
Conclusion & Perspectives

Across previous chapters, we have addressed many aspects of medical simulation, through a variety of solutions and applications. Our contributions to this field, among many others, have helped improve the realism of simulations, but more importantly they also contributed to a change in mindset in Medicine. Only ten years after the first attempts at designing virtual reality systems for procedural skills training in minimally invasive surgery, we start to see the first changes in training paradigms, in several areas of medicine. Thanks to the development of commercial systems, often inspired by academic work, training on a computer rather than on a patient has become a reality. In several instances it has even been demonstrated that simulation was as effective as conventional training, without putting patients at risk.

In spite of the incredible progress made during the past decade, several challenges remain. One of them consists in improving the realism of simulation systems, and extending simulation to all areas of medicine where computer-based training is possible. This will require the development of new approaches, maybe based on hybrid designs involving a more important hardware component. New models and new algorithms will need to be developed too, and as their complexity increases, they will have to take advantage of new generations of multi-core processors. Another key challenge is the development of medical simulation as an element of personalized medicine. In particular, simulations based on patient-specific data could be efficient tools for training and planning procedures. To reach these different goals, improvements need to be made in many directions. Anatomical modeling will obviously take a larger part in patient-specific simulations. Biomechanical modeling will also need to evolve considerably to allow for more realistic simulations and to account for patient-specific biomechanical parameters.

To pursue these research directions, with the long-term goal of developing realistic patient-specific simulations, some changes in our research and development approach are necessary. First, the multidisciplinary aspect of this research requires to develop collaborations and leverage everyone's expertise. Second, the various elements of this collaborative research need to be integrated into a software framework that will facilitate the integration of new contributions, that will allow validation of the results, and will simplify the development of applications based on this research. The following sections describe the main objectives and organization of a national research initiative we are developing across research teams at INRIA and in other institutions. In parallel with these research activities, we will pursue the development of the SOFA framework to support this important research effort and develop a series of applications in various clinical domains.

National Initiative on Medical Simulation

To further develop collaborations among experts in areas relevant to medical simulation, we propose to expand the current group of researchers involved in the development of the SOFA framework to create a stronger, research-oriented collaboration across multiple INRIA teams. A "national initiative" offers a structure that is well adapted to the multidisciplinary nature of medical simulation research. It also provides a collaborative framework that is well adapted to the integration of clinical and industrial partners, essential to the future creation of an international consortium of academic, clinical and industrial

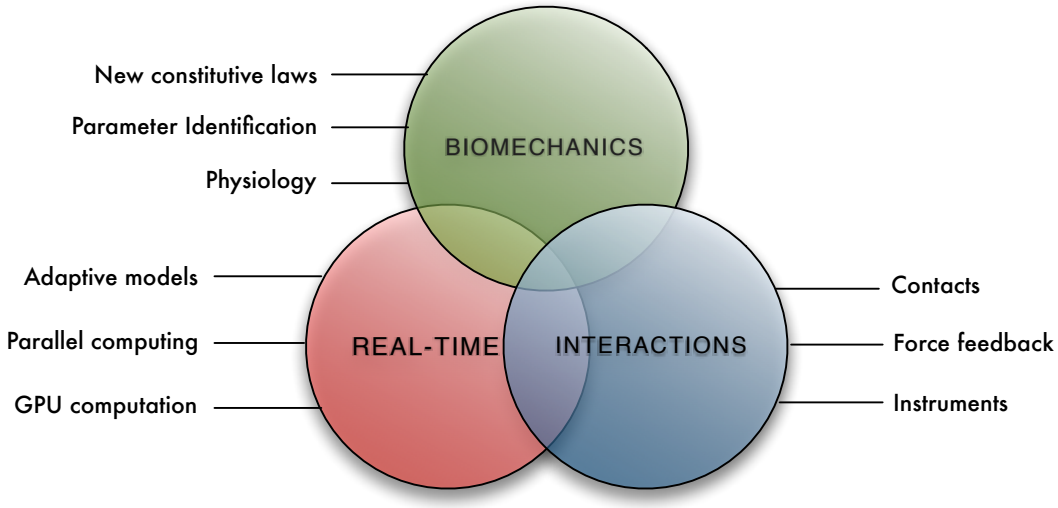
partners. To this end, the agenda proposed for the national initiative is geared towards three main objectives:

- Research and development of new algorithms dedicated to medical simulation
- Continuing development of the SOFA framework
- Development of a series of medical simulators as proofs of concept of our research activities

Although the primary objective of the national initiative is to strengthen research activities in the field of medical simulation, the existence of the SOFA framework will simplify the integration of the newly developed algorithms while offering a mean to compare and validate results.

Research Themes

This section describes the main research activities proposed for the national initiative. The diagram below illustrates the main research themes we plan to study and shows the interdependency between these themes, thus reinforcing the idea that only a coordinated program can significantly advance the state-of-the art.



Biomechanical models

Soft tissue modeling holds a very important place in Medical Simulation. A large part of the realism of a simulation, in particular for surgery or laparoscopy simulation, relies upon the ability to describe soft tissue response during tissue manipulation. Although several approaches have been proposed over the past ten years to model soft-tissue deformation in real-time (mainly for solid organs), these models have important limitations. The tradeoff between real-time computation and accuracy has often led to *ad hoc* models, lacking biomedical “roots”. More recently, several approaches based on elasticity theory have been proposed, using a finite element approach to solve the equations, but a majority of these works remains limited to linearized constitutive laws. Although significant improvements in both realism and computational efficiency have been obtained using such models, their range of application remains limited. Developing more realistic models of soft tissues that remain compatible with real-time computation is an essential research direction for which only few solutions currently exist. Such advanced models will not only permit to increase the realism of future training systems, but they will act as a bridge

toward the development of patient-specific preoperative planning as well as augmented reality tools for the operating room.

From a clinical standpoint, we will focus on two types of procedures where biomechanical modeling will play an important role: interventional radiology and laparoscopic surgery. In particular we will focus on the modeling of vascularized anatomical structures. Examples include organs such as the liver but also the kidneys, as well as the brain. When a therapy needs to be applied on such structures, it is currently possible to perform a procedure surgically or using an endovascular approach. This requires to characterize and model the behavior of vessels (arteries and veins) as well as the behavior of soft tissue (in particular the parenchyma). Another challenge of this research will be to model the interactions between the vascular network and the parenchyma where it is embedded. These interactions are key for both laparoscopic surgery and interventional radiology as they allow to describe the motion of the vessels in a vascularized organ during the procedure. This motion is either induced by the surgical manipulation of the parenchymal tissue during surgery or by respiratory, cardiac or patient motion during interventional radiology procedures.

Biomechanical modeling of vascular structures: the arterial wall is composed of three layers with very different purposes and characteristics. Because the artery wall is a complex living tissue with many fibers it cannot be treated with a simple linear response. Instead, a hyper-elastic model, which takes into account the different properties of the collagen and elastin fibers, is most likely to provide a realistic behavior. Such a representation will be combined with a finite element approach using shell elements. This will permit to describe changes in behavior, as well as rupture points during the development of an aneurysm of stenosis. This will be particularly relevant for planning coil embolization or stent placement.

Biomechanical modeling of parenchyma: the parenchyma is the main constituent of several important anatomical structures such as the brain, the liver, the kidneys, etc. Several recent studies have shown that this type of soft tissue has a viscoelastic or hyper-elastic behavior and that the capillary system plays an essential role in the time-dependent response of the tissue. When a large deformation is applied the response depends on the redistribution of the blood in the vascular system, particularly in the capillaries.

Soft-tissue characterization: an important step toward patient-specific planning is the integration of geometrical models of the patient's anatomy in the simulation. However, pathologies cannot be simply characterized by the shape of the organ. Information about tissue stiffness, or change in behavior due to the pathology, also need to be identified. There is very little work done in this area, in particular when tissue properties need to be measured *in vivo* and non invasively. Newly developed techniques, such as elastography, could be used to this end. Elastography is an imaging technique whereby local tissue strains are estimated from differential ultrasonic speckle displacements. These displacements are generated by a weak, quasi-static stress field and recorded in a strain image called an elastogram. During this program we will investigate different alternatives for measuring tissue properties *in vivo*, with a particular emphasis towards Magnetic Resonance Elastography. Inverse analysis will be performed to determine the parameters of a given constitutive model from the information measured on the patient.

Topological changes

Topological changes hold an important place in medical simulation, relating to many surgical procedures whether they involve soft-tissue cutting, tearing, blunt dissection or bone drilling. These aspects are complex to model as they require both geometrical and biomechanical descriptions of the local change in the anatomical structure. When real-time constraints also need to be accounted for, modeling topological

changes becomes a real bottleneck of current simulation systems. Our research in this area will focus on three main directions. The first step toward handling topological changes in real-time consists in defining **data structures** that allow very efficient access and modification of the structure of a mesh. Although such structures have been well studied for triangulations, it is not the case for tetrahedral meshes, hexahedral meshes as well as hierarchical or unstructured meshes. A second research direction will consist in defining **topological and geometrical operations** that will enable local and global re-meshing in real-time. This is a particularly difficult task when the newly created mesh needs to meet additional criteria (e.g. conforming mesh, or 3-manifold). In the field of Finite Element Modeling the domain of interest is traditionally described as a mesh composed of various types of elements (tetrahedral and hexahedral elements are the most common). Recently, **meshless techniques** i.e. that are not relying on an explicit mesh, have been introduced and could have promising applications in the area of medical simulation. We will investigate such approaches as they would simplify the management of topological changes.

Parallel and GPU computing

Although the past decade has seen a significant increase in complexity and performance of the algorithms used in medical simulation, major improvements are still required to enable patient-specific simulation and planning. Using parallel architectures to push the complexity of simulated environments further is clearly an approach to consider. However, interactive simulations introduces new constraints and evaluation criteria, such as latencies, multiple update frequencies and dynamic adaptation of precision levels, which require further investigation. We will particularly focus of the following aspects. **New parallel architectures**: parallel architectures, such as multi-cores CPUs, are now ubiquitous as the performances achieved by sequential units (single core CPUs) stopped to regularly improve. At the same time, graphical processors (GPU) offer a massive computing power that is more and more accessible to non-graphical tasks thanks to new general-purposes API such as CUDA from NVIDIA and CTM from AMD/ATI. GPUs are internally parallel processors, exploiting hundreds of computing units. These architectures could be exploited for more ambitious simulations, both on a single computer, and when considering clusters. A first prototype of GPU-based simulation developed in SOFA permitted to reach a simulation frequency 16 times faster than with a classical CPU. Such improvements permit to consider simulations with finer meshes for the simulated objects, or new algorithms modeling biomechanical behaviors more precisely. However, while the fast evolution of parallel architectures is useful to increase the realism of simulations, their variety (multi-core CPUs, GPUs, clusters, grids) make the design of parallel algorithm challenging. An important effort needs to be made to minimize the dependency between simulation algorithms and hardware architectures. **Adaptive and multilevel computations**: the performance improvements we can expect using recent hardware might not be sufficient to overcome the large increase of realism required in medical simulations. For this goal it is important to develop new algorithms that will better exploit these new architectures. A direction that we are privileging is to combine multi-level representations and locally adaptive meshes. Multi-level algorithms are useful not only to speedup computations, but also to describe different characteristics of the deformation at each level. Combined with local change of details of the mesh (possibly using hierarchical structures), the simulation can reach a high level of scalability.

We propose to study these approaches, with the additional objective of developing a generic strategy, allowing to coherently allocate resources between simulated elements, such that a certain level of performance is guaranteed while maximizing the realism. This problem, while similar to graphical level-of-detail approaches, is challenging because of the indirect relation between local precision changes and

their impact on the global simulation (a local phenomena can have important consequences at another point in time and space). Previous works on hierarchical simulation done by the ALCOVE and EVASION teams, and work on distributed simulations using multi-criteria adaptive scheduling, will serve as a starting point for this research.

Collision detection and contact modeling

As we have seen throughout previous chapters, collision detection, and particularly collision response, are two aspects of medical simulation that are often oversimplified. Yet, most medical procedures rely on interactions between a device and the anatomy, therefore involving many instances of complex contacts. Current approaches often rely on a simple contact point, rather than an actual area of contact, and rarely account for friction. While this simplification can produce plausible results in the case of an interaction between the end effector of a laparoscopic instrument and the surface of an organ, it is generally an incorrect approximation. As we move towards simulations for planning or rehearsal, accurately modeling contacts will take an increasingly important place. As in our previous work, we will continue to study contacts in the context of **interventional radiology** where the choice of the correct device has a direct impact on the development as well as the outcome of the procedure. Simulations on patient-specific data could assist in this choice. To reach this objective, it is key to model devices such as catheters, guidewires or coils, but also the interactions of these devices with the anatomy, using appropriate contact and friction laws. This is a particularly difficult problem given the large number of contacts to handle. Also, from a collision detection standpoint, **self-collisions** (that occur frequently during the deployment of a coil for instance) will need to be handled. Research activities regarding contacts will extend our current work on patient-specific planning of aneurysm embolization (in collaboration with the INRIA MAGRIT team). In **laparoscopic surgery**, the main challenge lies in the modeling of interactions between anatomical structures rather than between the instruments and the surface of an organ. During the different steps of a procedure organs slides against each other, while respiratory, cardiac and patient motion also generate contacts. Modeling these multiple interactions becomes even more complex when different biomechanical models are used to characterize the various soft tissues of the anatomy. Consequently, our objective is to accurately model resting contacts with friction, in a heterogeneous environment (spring-mass models, finite element models, particle systems, rigid objects, etc.). When different time integration strategies are used, a challenge lies in the computation of contact forces in a way that integrity and stability of the overall simulation are maintained.

Physiological modeling

Beyond biomechanical modeling of soft tissues, an essential component of a simulation is the modeling of the functional interactions occurring between the different elements of the anatomy. This involves for instance modeling physiological flows (blood flow, air flow within the lungs) or to model the biomechanical and physiological effect of pathologies, such as tumor growth and its impact on the surrounding tissues. Physiological modeling can also include muscular activity, in particular the coupling between electro-physiological and mechanical activities. We particularly plan to study the problem of **fluid flow** in two different contexts: the simulation of three-dimensional turbulent flow around aneurysms to better model coil embolization procedures, and the realistic simulation of bleeding during laparoscopic surgery. More specifically, we plan to use our current results on real-time laminar blood flow in complex vascular network to determine time-varying boundary conditions for a local, more detailed, flow model. A more prospective direction will be in the area of global physiology, i.e. models of the **systemic physiology** of a human body, able to predict functional changes such as blood pressure variation

resulting from blood loss, and generally able to model complications. Complications, are an essential element of a training curriculum. However, although complications contain elements of randomness that can be scripted in training scenarios, they also need to be modeled and fully integrated within the simulation loop. Finally, we also plan to adapt results of the CardioSense3D[8] project to determine, in real-time, the impact of **cardiac radio-frequency ablation**. This work would rely on our results in the field of interventional radiology simulation, but would extend its scope to patient-specific rehearsal of radio-frequency ablation. In particular we believe it would be possible to simulate the entire procedure, from catheter navigation and diagnostic to the actual therapy and the change in cardiac rhythm resulting from the delivery of the therapy.

Anatomical modeling

Image processing and three-dimensional reconstruction techniques required for anatomical modeling do not usually take into consideration requirements specific to interactive simulation. We propose to continue our work in developing a more **integrated reconstruction process** taking simulation constraints into account at various stages of anatomical modeling. This will permit to generate representations of the anatomy adapted to the different computational requirements of a simulation. Additionally, in the context of SOFA and to facilitate comparison and exchange of data across academic institutions, a reflection on the definition of a **standard for the description of three-dimensional anatomy** is necessary. While a standard exists for the description of medical images (DICOM), there is nothing similar to describe anatomical models. Some proposals such as MedX3D (from the Web3D consortium) offer only a limited description of the information required for medical simulations. In particular, biomechanical information, and high-level descriptions (semantics, topology, etc.) would be needed.

Validation tools

In Chapter 4, we have addressed the importance of validation at different levels of a simulation, from the validation of computational models to the assessment of the simulator as a training system. This is certainly an area where a lot of ground remains to be covered, and also where it is important to reach a consensus across academic groups that such validations are important. We will continue to work in this direction, mainly through the development of SOFA by proposing metrics and methodologies allowing easy assessments of newly developed algorithms.

It is important to mention here again that these directions for future work will not be addressed by one team only but by a group of teams working collaboratively and using SOFA as an integration framework. The duration envisioned for the national initiative is four years, and we hope to obtain significant results in some of these areas and to illustrate them through a series of prototypes of simulation systems. Needless to say that more than these four years will be necessary to make significant changes in all areas of medical simulation. Instead, we anticipate that we will obtain important preliminary results in these areas to initiate innovative, novel research directions and ideas. Using SOFA as the backbone for this research will help coordinate the different efforts with other research groups, and will offer a mean to assess our results.

[8] http://www-sop.inria.fr/CardioSense3D

Final words

To conclude this long list of perspectives, I think it is important not to forget why we are doing this work in the first place. Medical simulation should first and foremost be a tool to help improve Medicine and serve physicians. As such, we should always collaborate with those who will ultimately be our end users and who can guide us, to make sure that the solutions we propose are not only innovative and elegant, but that they also serve a purpose. To illustrate these important principles, I took the liberty to copy the text from two editorial comments that were written about our work. The first one was in response to an article in Annals of Surgery, one of the most prestigious medical journals, describing the research done during my Ph.D thesis. The second one commented on our article in "Catheterization and Cardiovascular Interventions", one of the leading journals in cardiology, describing our work in interventional radiology.

Surgical Simulation and Virtual Reality: The Coming Revolution

In the past decade, the world has accelerated its transition from the Industrial Age to the Information Age. Around your office or home, this revolution is apparent everywhere, from your digital watch to your personal computer, in the video games that your children (or you) play, and the enormous control that computers exert on utilities, transportation, banking and finance, and news and information. The relentless pace has, thus far, validated "Moore's Law" first described by Intel Corporations cofounder Gordon Moore, who observed that the power of computer chips doubles every 18 to 24 months. Randall Tobias, former vice-president of ATT, put it this way: "Over the last 30 years, we have seen a 3,000-fold improvement in computing power. If we had had similar progress in the automotive industry, a Lexus would cost $2, it would travel at the speed of sound, and go 600 miles on a thimble full of gas." In many aspects of life, computing power has been used beyond operations and computations, to run complex simulations and to create interactive virtual worlds. Predicting consumer behavior is currently being tackled by creating a data base of virtual people as unpredictable as real people.' Virtual shopping has the potential to transform the retail world. The U. S. Department of Defense has determined, based on extensive experience, that combat soldiers who survive their first 10 combat experiences have a 95% likelihood of surviving any further encounter. Thus, the military uses simulated or virtual reality combat training to produce those first 10 combat experiences in a non-lethal environment. The aviation industry has achieved an impressive safety record through a process of standardized education, repetitive practice, and frequent proficiency testing using aircraft simulators. Visualization software enables potential customers to operate an overhead crane in a virtual steel mill, or judge the ergonomics of a proposed front-end loader. While far from perfect, the progress has been substantial and the pace relentless. This same progress, so evident elsewhere in the world, has been less rapidly adopted in medical practice and education. The convulsive transformation of medicine from the rules of a cottage industry to those of a trillion-dollar business will, however, force similar applications. Beyond the simple computerization of medical records, information technology can and will be applied to the practice of medicine and surgery and our educational programs.

In a landmark paper in this issue of Annals of Surgery, Marescaux and colleagues, a unique collaborative group of surgeons and computer scientists, present their work on the application of virtual reality to hepatic surgery and simulation. The Revolution is here. These authors make a compelling case for the use of virtual reality, a unique form of computer-driven simulation, for surgical education, preoperative planning and practice, and the possibility that such information might ultimately be optimized and then delivered with robotic assistance. Each of these points are unique and worth considering individually, but first a few definitions are in order. Simulation can be

defined as a device or exercise that enables the participant to reproduce or represent, under test conditions, phenomena that are likely to occur in actual performance. There must be sufficient realism to suspend the disbelief of the participant. Virtual reality is an advanced human-computer interface that goes far beyond existing interfaces (e.g., a keyboard, mouse, or joy stick). Virtual reality implies a three-dimensional computer generated world that mimics the real world and allows participants to interact with and navigate it, using components of their five senses in real time, and become immersed. Haptics is the study of the psychomotor experience of touch, and is an essential component if a meaningful surgical experience is to be created.

The authors have incorporated a three-dimensional reconstruction of the sliced image data from the Visible Human project, sponsored by the National Library of Medicine. This project created the one-millimeter sectional anatomy used to build the three-dimensional liver model, and demonstrates the power of the Visible Human project. Using simultaneous processing and graphics workstations (currents costs: less than $100,000.), a projectable liver surface model was created as a wire-mesh construct. Photographs taken during laparoscopic procedures were then "texture mapped" over this surface to create a model that has the visible and haptic characteristics of the liver, and is interactively palpable using a commercially available force feedback device, the Impulse Engine. Such a reconstruction is a formidable task, but the authors have demonstrated the ability to move from the total-body data set of the Visible Human to a specific organ construction — the first step toward building other interactive surgical models. Using novel mathematic simplifications, a palpable and deformable organ has been created. **Historically, it has been said that any computer-generated virtual world can possess any two of three properties: good, fast, and cheap. Marescaux and colleagues have demonstrated that all three are possible! In addition, they have thoughtfully constructed a model which allows for the insertion of tumor masses, creating novel educational opportunities.**

Preoperative surgical planning has changed enormously, as imaging technology has advanced; intra-operative imaging has also refined planning. Heretofore, however, the ability to practice a surgical procedure preoperatively has been inconceivable. This report demonstrates that this is now possible, which suggests that a surgical procedure could be practiced on an individual patient's specific data. Put another way, Mrs. Smith's liver tumor could be virtually resected the night before her operation, based on her CT scan. While historically many surgical procedures were devised and practiced on animals before human application, virtual reality reconstruction theoretically allows totally patient-specific practice. **Of the five senses, the sense of touch is unique because it is interactive. When one looks at an object, the object does not look back, but when one lifts an object, an interaction immediately begins: the object is lifted with muscular force in direct proportion to the mass of the object and our intent. Thus, a five-pound object is lifted with a far different force than one would apply to a 50-pound object. Using a commercially available force feedback device, the Impulse Engine, which is used to simulate laparoscopic surgery, a palpable surgical manipulation is now possible. Real-time deformation is perceived in a surgical range of forces (25 g/meter with maximal forces of 8.9 Newtons), collision detection is done at the point of the tool, and deformation and resistance are updated 1,000 times per second for the force feedback and 50 images per second for the visual display. This represents true, real-time surgical virtual reality. The implications of this technology are enormous. Heretofore, our surgical education, at all levels, has remained locked in the age-old apprenticeship model, best described as "see one, do one, teach one." What this report suggests is the evolution of a surgical procedure curriculum. Basic surgical skills could be learned in a systematic and logical fashion by doing, and a student's or resident's proficiency could be safely assessed and demonstrated. Only after a demonstration of such skill and proficiency would those skills then be refined in the**

operating room. Such a concept has real potential for providing more effective, less expensive, and safer surgical education.

The third essential point that Marescaux and colleagues make is the ingenious and innovative suggestion that a pre-practiced surgical procedure might then be optimized as a stored data set of movements and replicated intra-operatively with robotic assistance. Consider the analogy of making a movie: in the film industry, the participants perform a given scene and repeat it until the director is satisfied; many scenes are then spliced together, edited, and optimized to produce the movie the director envisioned. Looking ahead with this technology, a surgical procedure could be rehearsed and stored as a computer data set. This data set could then be optimized, false moves eliminated, and the perfect surgical procedure delivered by computer-driven and computer-enhanced robotic arms. This is a bold projection, but it is by no means ludicrous — consider the current availability of the voice-controlled robotic arm to control the video camera during laparoscopic surgery.

Interest in the use of virtual reality in medicine in the United States has been supported by the Advanced Research Projects Agencies (ARPA) of the Department of Defense and championed by its former director, Col. Richard Satava, MD, now a professor of surgery at Yale. The annual Medicine Meets Virtual Reality Congress and Surgical Robotics Congresses, and the Technology Demonstration at this year's Clinical Congress of the American College of Surgeons, sponsored by the Committee on Emerging Surgical Technologies and Education, are clear evidence that the technology is compelling. The field of anesthesia has been an important early adopter in crisis resource management; basic science curricula have also considered virtual reality methodology. It is important to remember that all of these early efforts are very much the equivalent of the "Model T" of the automobiles that we take for granted today. Where will this all go? Dr. Harold Varmus, NIH Director, has announced plans to direct increased support toward accelerating bioengineering research. **It seems clear that unique partnerships between surgeons, engineers, and computer scientists, like the Marescaux group, will be needed to drive this field in the most appropriate and useful direction, and to demonstrate educational validity.** It does seem likely that future generations of surgeons will be selected, trained, credentialed, and recredentialed using simulation and virtual reality devices like the one discussed here and others yet to be developed. The mantra of the Information Age as it applies to surgery is probably best stated by Col. Satava: "It's no longer blood and guts, it's bits and bytes."

Thomas M. Krummel, MD
John A. and Marian T. Waldhausen Professor of Surgery
Chair, Department of Surgery
Penn State University College of Medicine, Pennsylvania, USA

Computerized Patient Simulation to Train the Next Generation of Interventional Cardiologists:

Can Virtual Reality Take the Place of Real Life?

The tradition of clinical teaching extant from Hippocrates and Osler has been concisely stated as “Watch one, do one, teach one.” The general concept is that the experiential facets of medicine, especially procedural techniques, are non-cognitive and thus can not be taught in a classroom but have to be learned at “the bedside” by first observing and then copying “the master.” In the catheterization laboratory, as with most surgically oriented disciplines, the practicality of training young physicians requires long and intensive apprenticeships with progressively extensive involvement.

Responsibility for decision making, however, resides fully with the attending physician until training is "completed," when the newly certified physician develops one's own "practice." In this classical model, institutions where training occurs are called "teaching hospitals," now usually affiliated with medical schools. The strengths and weaknesses of this approach have been depicted in television series, movies, fiction, and non-fictional accounts for generations in eloquent fashion ("Of Human Bondage", "House of God"). The basic, almost romantic, notion is that the trainee learns the trade by doing, making minor errors along the way that are quickly recognized through the close observation of the teaching physician with "instant" correction. The neoclassical interpretation, of course, is that we all know that the errors aren't always caught but the patients all do well despite our, or our trainee's, mistakes—leading to a sense of fraternity in a shared experience. The inherent risks to the patient almost never materialize, sometimes through skill, other times by luck or serendipity.

The problem is that this description of medical education is not now and never was quite true — patients have been harmed by trainees for centuries. Furthermore, physicians continue to learn and improve technical skills throughout their career; certification gives one a license to be a doctor, but does not signify the end of learning new things. Just as every patient is different, so is every intervention, each with its own challenges and risks. Still, with no better method to teach and continue to learn, the masquerade is best left unperturbed.

But suppose that there was another way to learn and teach that did not take risks with our patients and gave more experienced operators a way to maintain their skills in a hands-on setting? We expect airline pilots who operate commercial flights to be fully trained once they are in the cockpit, and the analogy to flight simulators is quite apt. If a computer could be programmed to simulate cath-lab conditions that reproduce the hand-eye coordination, manual dexterity, and judgmental richness of coronary interventional procedures, the implication would be enormous. **Beyond basic training for the cardiology and interventional fellows, more advanced train- ing could be developed for the newer physician. Specific scenarios featuring the use of a certain device or a common clinical challenge (i.e., "kissing balloons," "vein graft interventions") could be taught and mastered before dealing with an actual case in the cath-lab.**

Those of us who took interventional boards last year and were not impressed that the test measured competency can be especially intrigued by the possibility of a simulator used as a certification tool. Returning to the pilot analogy, flight scenarios are built to recreate various catastrophes or potential problems to teach the pilot how to respond, so that if the same occurrence should happen in "real-life," there is some background and rudimentary experience to draw upon. The same could be done in our field, where each of us can recall our "nightmare" cases that led to deeply emotional experiences. Additionally, the question of operator volume may be made less crucial by this tool.

It is in this context that the report by Dawson *et al.* is best understood. The methodology contained in their report may well represent a breakthrough of astonishing and revolutionary importance. I hope it is the beginning of a new chapter in medical training that exposes many of our profession's strongest shibboleths. Certainly, virtual reality is not quite as good a teacher as the real thing, except that no one actually suffers.

Lloyd W. Klein MD, FSCAI
Rush Heart Institute, Rush Presbyterian St. Lukes
Medical Center, Chicago, Illinois, USA

SYNTHESIS OF ACTIVITIES

Resume

EXPERIENCE

RESEARCH DIRECTOR, INRIA, LILLE, FRANCE — 2007 -

I joined INRIA in February 2007 to hold a position of Research Director. My principal research activities continue to revolve around the simulation of medical procedures, for training and planning purposes. This involves various aspects such as biomechanical modeling of soft tissue, mechanical modeling of flexible medical devices such as stents, catheters and guide-wires. I am also involved in research on contact modeling between deformable structures, performance metrics for training systems, and patient-specific anatomical modeling adapted to real-time simulation constraints. I am also responsible for the development of a national initiative on Medical Simulation using the SOFA framework (www.sofa-framework.org) as a common platform for research, integration and validation of new algorithms.

RESEARCH LEAD, SIM GROUP, CIMIT, CAMBRIDGE, USA — 1998 - 2007
INSTRUCTOR, HARVARD MEDICAL SCHOOL, BOSTON, USA — 1999 - 2007

For seven years I have acted as Research Lead for the Medical Simulation Group at CIMIT in Boston where I was responsible for defining research directions and technical infrastructures for several simulation projects, including a Chest Trauma Training System, a Computer-Enhanced Laparoscopic Training System and an Interventional Radiology Training System.

RESEARCH SCIENTIST, MITSUBISHI ELECTRIC RESEARCH LAB, CAMBRIDGE, USA — 1998-1999

For more than a year I was involved with a spin-off of Mitsubishi Electric Research Lab. This group of five people worked on the development and commercialization of an interventional cardiology training system. The project, called ICTS was transfered to Mentice and is now commercialized as the VIST system.

EDUCATION

1991 — B.S., University of Burgundy, France

1992 — M.S., University of Burgundy, France

1997 — Ph.D., University of Nice - Sophia Antipolis, France

SCIENTIFIC ACTIVITIES

SCIENTIFIC BOARDS

2008 — Member of the Program Committee of the MICCAI conference

2008 — Member of the Steering Committee of the International Symposium on Computational Models for Biomedical Simulation symposium

2007 — Member of the Scientific Board of the VRIPHYS conference

2006 — Member of the Steering Committee of the International Symposium on Computational Models for Biomedical Simulation symposium

2006 — Member of the Scientific Board of the VRIPHYS conference

2005 — Member of the Steering Committee of the International Symposium on Computational Models for Biomedical Simulation symposium

2005 — Member of the Scientific Board of the VRIPHYS conference

2004 — Organizer and editor of the International Symposium on Medical Simulation symposium

2003 — Member of the Scientific Board of the 3rd International Symposium of Surgery Simulation and Soft Tissue Modeling symposium

2003 — Meeting Chair, International Symposium of Surgery Simulation and Soft Tissue Modeling

SCIENTIFIC REVIEWS

I am a regular reviewer for the following international conferences or journals: International Symposium on Medical Simulation, MICCAI, VRIPHYS, IEEE Transactions on Visualization and Computer Graphics, IEEE Transactions on Medical Imaging, and Media (Medical Image Analysis).

PROFESSIONAL SOCIETIES

I am a member of the Society for Medical Simulation since 2005.

AWARDS AND HONORS

2006 — Poster of Distinction Award housa– 30th Annual MGH Research Symposium

2005 — US Army's Greatest Inventions Award

2005 — Best Paper Award – VRIPHYS'05 conference, Pisa, Italy

2003 — The Edward M. Kennedy Award for Health Care Innovation

1997 — Ph.D. thesis, graduated Summa Cum Laude

SELECTED PUBLICATIONS

INTERNATIONAL PEER-REVIEWED CONFERENCES

Delingette H, **Cotin** S and Ayache N, A Hybrid Elastic Model Allowing Real-Time Cutting, Deformations and Force-Feedback for Surgery Training and Simulation, Computer Animation, pp. 70-81, 1999.

Cotin S, Stylopoulos N, Ottensmeyer M, Neumann P, Rattner D, Dawson S. Metrics for Laparoscopic Skills Trainers: The Weakest Link! Dohi T and Kikinis R, eds. Proceedings of MICCAI 2002, Lecture Notes in Computer Science 2488, pp. 35-43, Springer-Verlag; 2002

Luboz V, Wu X, Krissian K, Westin C.F, Kikinis R, **Cotin** S, Dawson S. A segmentation and reconstruction technique for 3D vascular structures. Proceedings of the MICCAI Conference, MICCAI, pp. 43-50, 2005.

Cotin S, Duriez C, Lenoir J, Neumann P, Dawson S. New approaches to catheter navigation for interventional radiology simulation. Proceedings of the MICCAI Conference, MICCAI 2005, pp. 534-542, 2005.

Dequidt J., Lenoir J., **Cotin** S. Interactive Contacts Resolution Using Smooth Surface Representation. Proceedings of the MICCAI Conference, LNCS 4792, pp. 850-857, 2007.

Wu X., Allard, J., **Cotin** S. Real-time Modeling of Vascular Flow for Angiography Simulation. Proceedings of the MICCAI Conference, LNCS 4792, pp. 850-857, 2007.

INTERNATIONAL JOURNALS

Cotin S, Delingette H, Ayache N. Real-time elastic deformations of soft tissue for surgery simulation. IEEE Transactions on Visualization and Computer Graphics 1999; 5(1): 62-73.

Dawson S, **Cotin** S, Meglan D, Shaffer D, Ferrell, M. Designing a Computer-Based Simulator for Interventional Cardiology Training. Catheterization and Cardiovascular Interventions 2000; 51:522–527.

Cotin S, Delingette H, and Ayache N. A Hybrid Elastic Model allowing Real-Time Cutting, Deformations and Force-Feedback for Surgery Training and Simulation. The Visual Computer, 16(8): 437-452, 2000.

Kerdok A, **Cotin** S, Ottensmeyer M, Galea A, Howe R, Dawson S. Truth Cube: Establishing Physical Standards for Real Time Soft Tissue Simulation. Medical Image Analysis 2003; 7:283-291.

Duriez C, **Cotin**, S and Lenoir, J. New Approaches to Catheter Navigation for Interventional Radiology Simulation, in: Computer Aided Surgery, 2006, vol. 11, p. 300-308.

PATENTS AND PATENTS APPLICATIONS

Cotin S, Delingette H, Ayache N inventors; “Electronic Device for processing image-data for simulating the behavior of a deformable object”, International patent WO9926119A1, 1998

Dawson S, **Cotin** S, Ottensmeyer M, Neumann P inventors; “Medical Training System”, United States Patent Application, 10/488,415 - United States, Canadian, and European patent application PCT/US02/28593 – March 2, 2004.

Cotin S, Stylopoulos N, Dawson S, Ottensmeyer M inventors; “Surgical Training System for Laparoscopic Procedures”, United States Patent Application, 10/797,874 – March 10, 2004.

Cotin S, Bardsley R, Dawson S inventors; “Design of a new haptic/tracking interface for interventional radiology training”, Massachusetts General Hospital Invention Disclosure

Cotin S, Wu X, Neumann P, Dawson S inventors; “Methods and Apparatus for Simulation of Endovascular and Endoluminal Procedures”, United States Patent Application, 60/600,188 - PCT application number PCT/US2005/028594, August 10, 2005

Short Biography

From 1998 to 1999 I worked as research scientist at Mitsubishi Electric Research Laboratories in Cambridge, Massachusetts, as one of the five team members of a spin-off group. Our goal was to develop a computer-based training system for interventional cardiology, permitting medical procedures to be learned in a situation free of risk to patients. My main role on this project was to develop a real-time vascular flow model as well as a method for computing the advection-diffusion characteristics of contrast agent mixing with blood. Besides the research aspects of this project, we also developed a business plan and studied commercialization strategies, as our end goal was to start up a company. Since, the rights to this simulator were purchased by Guidant Corporation, and later by Mentice Corporation (www.mentice.com). The system, now named VIST, has been installed worldwide and used to train several thousands of interventional cardiologists in coronary angiography, coronary angioplasty, coronary stenting and dual-lead pacemaker placement. In 1999, I co-founded the SimGroup, a research team based at CIMIT, in Boston. As Research Lead for this group, one of my primary goals was to define new research directions for the group or extend the scope of our work in areas already established as our core science: three-dimensional anatomical modeling, soft tissue modeling and medical device modeling, tissue-tool interactions, tracking and haptic devices, and performance metrics. Another part of my activities consisted in defining clinically-relevant applications that can build upon our core science. This approach led to several successful projects and advances in various research areas. Among the most successful applications of this research were a chest trauma training system (VIRGIL), a laparoscopic training system (CELTS), and a training system for neuro-interventional radiology (EVE). This research has also been the source of publications in key conferences and journals, and has been the subject of feature articles in local and national newspapers and various TV programs (7 News, Channel 5, the SciFi channel). Several of our research projects have also drawn interest from companies and are under discussion for potential commercialization. Our two most important research projects involved research on soft tissue properties and modeling, and simulation techniques for interventional radiology. The first one, entitled Enabling Technologies for Advanced Soft Tissue Modeling aimed at proposing a suite of instruments and models for describing soft biological tissues. Accomplishments include: development of a detailed model describing the poroviscous, hyperelastic character of tissues such as parenchyma ; development of inverse techniques for the extraction of the characteristic parameters from this model; integration of a three-dimensional ultrasound scanning apparatus to measure 3-D strain fields during indentation. The second project, Simulation of Interventional Radiology Procedures, targeted stroke therapy as a first step toward general interventional radiology simulation. The research aspects of this project involved medical image processing for three-dimensional reconstruction of vascular networks, new compact multi-level representations for vascular networks, real-time finite element modeling of flexible medical devices, collision response strategies for highly constrained deformable objects, fluroroscopic image synthesis, and the design of a new haptic interface. A large part of my research was done under the umbrella of the Telemedicine and Advanced Technology Research Center Medical Simulation and Training Initiative (MSTI). A large part of the work presented in this manuscript was funded by CIMIT and the US Department of Defense, and was done in collaboration with the Harvard University Division of Engineering and Applied Science, the MIT Laboratory of Computer Science, the Department of Surgery and the Department of Radiology at Massachusetts General Hospital.

Scientific Responsibilities

From 1999 to 2007 I assumed the role of research lead of the Sim Group at CIMIT, where I managed a team of several permanent researchers and post-doctoral fellows. My role involved many different aspects, but principally consisted in understanding and analyzing the needs on the clinical side and translating these needs into requirements for our research program. It also involved recruiting researchers to develop our research activities in various domains, and launching several prospective research initiatives. I also participated in numerous meetings with representatives of commercial entities interested in licensing our research.

I am an active reviewer for the following journals and conferences: Journal of Computer Vision and Image Understanding, CVRMed and VBC conferences, Media (Medical Image Analysis), International Symposium on Computational Models for Biomedical Simulation, IEEE Transactions on Visualization and Computer Graphics, Virtual Reality Interactions and Physical Simulations (VRIPHYS), and Graphical Models. In June 2004, I organized the third International Symposium on Medical Simulation, held in Boston. In January 2008, I was co-chair of a Technology Track at the International Meeting in Simulation in Healthcare (IMSH'08), the largest meeting dedicated to Medical Simulation.

I am also a member of the Society for Medical Simulation (www.socmedsim.org). I have participated to focus groups at the National Library of Medicine and the Telemedicine and Advanced Technology Research Center. I am also an active member of AIMS, the Advanced Initiatives in Medical Simulation, a coalition of individuals and organizations committed to promoting medical simulation by engaging, energizing, and further developing the medical simulation community and creating awareness and educating about simulation (www.medsim.org). This association organizes an annual meeting that includes a congressional exhibition where demonstrations are made to members of the US congress. I am or have been a member of the program committee or steering committee of the following conferences: International Symposium on Computational Models for Biomedical Simulation, Workshop in Virtual Reality Interactions and Physical Simulations, International Symposium of Surgery Simulation and Soft Tissue Modeling, International Symposium on Computational Models for Biomedical Simulation.

Supervision of Research Activities

Graduate Students

- 2000: Michal Berris, Mechanical Engineering, Ph.D. candidate, MIT - 9 months - soft tissue properties and haptic feedback for chest trauma simulator - 100% supervision
- 2001: Yann Chaumet, International Exchange Program, Master's student - 9 months - development of a Web and Java interface for C++ applications - 100% supervision
- 2002: Ryan Culberson, Master's student, Tufts University - 5 months - real-time collision detection based on octree partitioning and spatial coherence - 100% supervision
- 2003: Vincent Pegoraro, International Exchange Program, Master's student - 12 months - real-time simulation of fluoroscopic images and contrast agent rendering - 100% supervision

- 2005: Gontran Magnat, International Exchange Program, Master's student - 6 months - research on meshless techniques for real-time deformable models - 50% supervision

Postdoctoral Fellows

- 2002 – 2003: Manivannan Muniyandi, Ph.D. - 5 months - fast volume rendering using the two-dimensional multi-texturing - 100% supervision
- 2002 – 2003: Xunlei Wu, Ph.D. - 12 months - soft tissue modeling ; simulation of advection and diffusion effects in fluids ; surface reconstruction of vascular structures - 80% supervision
- 2003 – 2006: Vincent Luboz, Ph.D. - 26 months - soft tissue properties measurement ; development of phantom models ; segmentation and reconstrution techniques for vascular structures - 60% supervision
- 2005 – 2006: Christian Duriez, Ph.D. - 11 months - contact modeling for highly constrained flexible models ; real-time deformation of wire-like structures - 80% supervision
- 2005 – 2007: Julien Lenoir, Ph.D. - 20 months - real-time model of coils based on dynamic 1D splines ; advanced visualization techniques for interventional radiology simulation - 80% supervision
- 2006 – 2007: Jeremie Allard, Ph.D. - 12 months – development of the SOFA framework and integration of GPU-based parallel computation techniques; advanced rendering and physiological modeling for interventional radiology simulation

Research Fellows

- 2001 – 2003: Nicholas Stylopoulos, M.D. - 18 months - performance metrics for laparoscopic skills training ; setup and conduct of validation studies - 80% supervision

Collaborations

Guidance System for Hip Surgery: project developed in collaboration with for Cambridge Medical Devices, LLC and Ascension Technologies Inc. My role was to serve essentially as a consultant on most aspects of the system as well as a developer of an initial prototype.

Training System for Epidural Catheter Insertion: project developed in collaboration with Physical Sciences Inc. as part of a SBIR award from the National Institutes of Health. My role is to act as a consultant on different aspects of the development of the system with a particular emphasis on the design of performance metrics and well as an augmented reality environment.

Simulation Open Framework Architecture: the SOFA project, initiated while I was at CIMIT, became quickly of interest for researchers from INRIA. We initiated a collaboration with the ALCOVE, ASCLEPIOS and EVASION teams from INRIA to further develop the project. This led to a very active and fruitful collaboration, with for instance the release of the first public version of SOFA in February 2007, and a couple of publications with authors from both INRIA and CIMIT.

Simulation and Planning of Coil Deployment in Cerebral Aneurysms: a collaborative action (ARC) was started early 2007 between the ALCOVE team and the MAGRIT team at INRIA, as well as with the teaching hospital in Nancy. The objective of this two-year project is to develop an interactive planning system to guide interventional radiologist during the placement of coils inside aneurysms in the brain.

Dissemination of Scientific Knowledge

In 1996 and 1997, as I was Ph.D. candidate at INRIA in France, my work on laparoscopic surgery simulation was featured at several occasions on national TV programs, including national news (France 3 in 1996), a national TV Health Program (Santé à la Une in 1997). It was also featured on a national radio program (France Info, 1997). I was also featured in VSD Magazine in a 3 page report on my research project. The article was entitled "Opérer Comme Dans Un Jeu Vidéo" (VSD, November 1996, pp. 26-28). Finally this work was presented at the French Academy of Science by Professor Jacques Marescaux. The work developed at CIMIT has been featured at several occasions in local and national newspapers: the Boston Herald, Boston Business Forward, and the New Yorker, as well as local and national TV programs (7 News, Channel 5, the SciFi channel).

Scientific Exchanges

Just after finishing my PhD, in 1998, I moved to Boston, USA to join a spin-off group of Mitsubishi Electric Research Lab, as a Research Scientist. In 1999, I co-founded the Simulation Group at CIMIT where I worked as Research Lead until May 2007. From October 2004 to January 2006, I spent half of my time as visiting scientist with the Alcove team in Lille. My main role there was to initiate the creation and development of an international project (SOFA) aimed at designing an Open Source software framework for medical simulation.

Software Development

Real-time Finite Element Model: code developed during my PhD thesis; development time: 30 months; more than 20,000 lines of code; used in the development of a prototype of a training system for laparoscopic surgery with force feedback; served as the basis for the PhD thesis of Guillaume Picinbono and for the work done through the collaborative initiative AISIM. This work also led to an international patent application.

Interventional Cardiology Training System: development time : 15 months; more than 20,000 lines of code; co-developers: 3 persons. This code served as the core for the development of a commercial product named VIST (www.mentice.com)

Chest Trauma Training System: simulator combining a real-size patient mannequin, electromagnetic position sensors, and a computer system. VIRGIL won the prestigious Army's Greatest Inventions Award in 2004. Initially designed for training combat medics, VIRGIL has also applications in emergency medicine. Development time: 12 months; about 10,000 lines of code; protected by US and European patents.

Computer Enhanced Laparoscopic Training System: implementation of a new concept for performance metrics and scoring based on kinematics analysis; development of a fully functional prototype that has passed several validation tests, and was ranked first in a validation study involving several existing commercial products. Development time: 18 months; unique developer; about 5,000 lines of code; protected by US patent; commercialization discussions initiated with a medical device company.

Guidance System for Hip Surgery: the code developed in this application, aimed at being easy to use and very robust, involved position tracking of medical devices, registration and visualization techniques. Work was done in collaboration with a start-up company, Cambridge Medical Devices, LLC. Development time: 24 months; unique developer; about 10,000 lines of code; system tested in clinical setting.

High-Fidelity Training System for Interventional Radiology: this project, which goal is to develop a very high-fidelity simulation system, is currently targeted at stroke therapy. However, the models and solutions introduced in this prototype can be applied to any interventional radiology specialty, from cardiology to abdominal interventions. Development time: 24 months; about 30,000 lines of code; co-developers: 5 persons ; protected by US patent.

SOFA - Simulation Open Framework Architecture: this project is an international, multi-institution collaborative initiative, aimed at developing an open source and extendible framework for Medical Simulation. The main objective of SOFA's research and development is to foster collaboration among research groups. Project still under developments; about 200,000 lines of code; co-developers: 8 persons. The SOFA project is, at the beginning of 2008, the 2nd most downloaded software on the INRIA gForge, with more than 20,000 downloads.

Technology Transfer

Most of the research I have been involved with since my PhD thesis has been integrated in various prototypes and has led to patent applications. Through my work at CIMIT I have also been involved in several industrial collaborations, and I have written several successful grant applications.

Research Grants and Awards

- $200,000 TATRC Award, "Simulation Open Framework Architecture", **Principal Investigator**, 2006
- $750,000 NIH SBIR Award, "Epidural Training System", co-investigator, 2005
- $100,000 CIMIT Award, "Computational Tools for Interventional Radiology", **Principal Investigator**, 2005
- $75,000 CIMIT Career Award, 2004 - **Principal Investigator**
- $100,000 CIMIT Award, "Hybrid Simulation System for Stroke Therapy Team Training", P.I., 2004
- $1,000,000 Department of Defense, "Enabling Technologies for Medical Simulation", co-investigator, 2002
- $1,000,000 Department of Defense, "Enabling Technologies for Medical Simulation", co-investigator, 2001
- $1,000,000 Department of Defense, "Enabling Technologies for Medical Simulation", co-investigator, 2000
- $945,000 Department of Defense, "Enabling Technologies for Medical Simulation", co-investigator, 1999
- $1,955,000 DHRP Award, "Advanced Soft Tissue Modeling", co-investigator, 2001 - 2005

Patents and Patent Applications

- **Cotin** S, Delingette H, Ayache N; "Electronic Device for processing image-data for simulating the behavior of a deformable object", International patent WO9926119A1, 1998
- Dawson S, **Cotin** S, Ottensmeyer M, Neumann P; "Medical Training System", United States Patent Application, 10/488,415 - United States and European patent application PCT/US02/28593, 2004.
- **Cotin** S, Stylopoulos N, Dawson S, Ottensmeyer M; "Surgical Training System for Laparoscopic Procedures", United States Patent Application, 10/797874, 2004.
- **Cotin** S, Bardsley R, Dawson S; "Design of a new haptic/tracking interface for interventional radiology training", Massachusetts General Hospital Invention Disclosure
- **Cotin** S, Wu X, Neumann P, Dawson S; "Methods and Apparatus for Simulation of Endovascular and Endoluminal Procedures", United States Patent Application, 60/600,188 - PCT application number PCT/US2005/028594, 2005.
- Luboz V, Wu X, Krissian K, **Cotin** S, inventors; “ADAM: a segmentation and reconstruction method for endovascular and endoluminal anatomical structures”, Provisional patent application, 2006.

Prizes and Awards

- **2005**: Best Paper Award VRIPHYS'05 conference, Pisa, Italy
- **2005**: Army's Greatest Inventions Award US Department of Defense, Washington DC
- **2004**: Simulators developed by the Sim Group at CIMIT chosen to be part of a permanent collection at the National Museum of Health and Medicine in Washington, DC.
- **2003**: Satava Award for our work in medical education and technology, MMVR Conference
- **2003**: The "Edward M. Kennedy Award" for Health Care Innovation Partners Health Care, Boston
- **1997**: Ph.D. thesis, graduated Summa Cum Laude University of Nice - Sophia Antipolis

Publications

Selected publications

Cotin S, Delingette H, Ayache N. “Real-time elastic deformations of soft tissue for surgery simulation”. IEEE Transactions on Visualization and Computer Graphics 1999 ; 5(1) : 62-73.

This article describes a real-time deformable model based on linear elasticity theory and a modified Finite Element Method. This journal article summarized some of the key contributions of my PhD thesis. The model presented in this paper was the first complex FEM-based model to be computed in real-time even for complex geometries. It was used for the development of a laparoscopic simulator for hepatic surgery. One of the illustrations of this work was used as the cover page of the journal.

Quatrehomme G, Cotin S, Subsol G, Delingette H, Garidel Y, Grevin G, Fidrich M, Bailet P and Ollier A. "A Fully Three-dimensional Method for Facial Reconstruction based on Deformable Models". Journal of Forensic Science 1997 ; 42(4) : 649 :652.

In this journal article we presented some very innovative work on facial reconstruction, i.e. the estimation of the shape of a human face from the shape of the skull and very limited anatomical information. The method, entirely manual until then, could potentially be replaced by a combination of registration techniques and deformable models. This work was done in collaboration with a forensic doctor, and published in the leading journal in the field.

Dawson S, Cotin S, Meglan D, Shaffer D, Ferrell, M. "Designing a Computer-Based Simulator for Interventional Cardiology Training". Catheterization and Cardiovascular Interventions 2000 ; 51 : 522-527.

This article, published in a prestigious journal, describes the project developed at Mitsubishi Electric. The main contribution of the paper is the description of a complete training system for interventional cardiology. The realism of the simulation, as well as its important applications, triggered an enthousiastic editorial comment, ending with the following remark: "The methodology contained in their report may well represent a breakthrough of astonishing and revolutionary importance. I hope it is the beginning of a new chapter in medical training that exposes many of our profession's strongest shibboleths."

Cotin S, Duriez C, Lenoir J, Neumann P, Dawson S. "New approaches to catheter navigation for interventional radiology simulation". Proceedings of MICCAI 2005, J. Duncan, G. Gerig (eds.), Lecture Notes in Computer Science, Volume 3749, pp 534-542, October 2005.

This article describes a complex real-time finite element model suited for wire-like structures, such as catheters, guidewires, coils or surgical threads. A method for handling the complex contacts between the wire and the surface of a vascular model is also introduced. The article was accepted as oral presentation at MICCAI, a leading conference in medical image computing and computer assisted interventions, with an acceptance rate of 7.4% for oral presentations. It was also selected from the best papers for publication in Computer Aided Surgery.

Additional important publications

- Dequidt J., Lenoir J., **Cotin** S. Interactive Contacts Resolution Using Smooth Surface Representation. Proceedings of the MICCAI Conference, LNCS 4792, pp. 850-857, 2007.
- Allard J, **Cotin** S, Faure F, Bensoussan P-J, Poyer F, Duriez C, Delingette H, and Grisoni L. SOFA – an Open Source Framework for Medical Simulation. In Medicine Meets Virtual Reality (MMVR), 2007.
- Luboz V, Wu X, Krissian K, Westin C.F, Kikinis R, **Cotin** S, Dawson S. A segmentation and reconstruction technique for 3D vascular structures. Proceedings of MICCAI 2005, J. Duncan, G. Gerig (eds.), Lecture Notes in Computer Science, Volume 3749, pp 43-50, October 2005.
- Kerdok A, **Cotin** S, Ottensmeyer M, Galea A, Howe R, Dawson S. Truth Cube : Establishing Physical Standards for Real Time Soft Tissue Simulation. Medical Image Analysis 2003 ; 7 :283-291.

- Marescaux J, Clement JM, Tassetti V, Koehl C, **Cotin** S, Russier Y, Mutter D, Delingette H, Ayache N. Virtual reality applied to hepatic surgery simulation : The next revolution. Annals of Surgery 1998, 228(5) :627-634.
- Bro-Nielsen M and **Cotin** S : Real-time volumetric deformable models for surgery simulation using nite elements and condensation, Eurographics'96, Computer Graphics Forum 1996 ; 15(3) :57-66.

Complete list of publications

Ph.D. Thesis

[1] "Real-time anatomical deformable models - Application to laparoscopic surgery simulation with force feedback", Ph.D. thesis, University of Nice - Sophia Antipolis, 1998.

Journal articles

[1] Duriez C, **Cotin**, S and Lenoir, J. New Approaches to Catheter Navigation for Interventional Radiology Simulation, in: Computer Aided Surgery, 2006, vol. 11, p. 300-308.

[2] Stylopoulos N, **Cotin** S, Maithel SK, Ottensmeyer M, Jackson PG, Bardsley RS, Neumann PF, Rattner DW, Dawson SL. Computer-enhanced laparoscopic training system (CELTS) : bridging the gap. Surg Endosc 18(5):782-789, 2004.

[3] Kerdok A, **Cotin** S, Ottensmeyer M, Galea A, Howe R, Dawson S. Truth Cube : Establishing Physical Standards for Real Time Soft Tissue Simulation. Medical Image Analysis; 7:283-291, 2003.

[4] Dawson S, **Cotin** S, Meglan D, Shaer D, Ferrell, M. Designing a Computer-Based Simulator for Interventional Cardiology Training. Catheterization and Cardiovascular Interventions; 51:522-527, 2000.

[5] **Cotin** S, Delingette H, Ayache N. Real-time elastic deformations of soft tissue for surgery simulation. IEEE Transactions on Visualization and Computer Graphics 1999 ; 5(1) : 62-73.

[6] Marescaux J, Clement JM, Tassetti V, Koehl C, **Cotin** S, Russier Y, Mutter D, Delingette H, Ayache N. Virtual reality applied to hepatic surgery simulation : The next revolution. Annals of Surgery 1998, 228(5) :627-634.

[7] Ayache N, **Cotin** S, Delingette H, Clement JM, Marescaux J, Nord M. Simulation of Endoscopic Surgery. Journal of Minimally Invasive Therapy and Allied Technologies 1998 ; 7(2) :71-77.

[8] **Cotin** S, Delingette H, Ayache N, Clement J.-M, Marescaux J, and Nord M. Simulation Active de Chirurgie Endoscopique. Revue Européenne de Technologie Biomédicale (RBM), 19(5) :167-172, 1997.

[9] Quatrehomme G, **Cotin** S, Subsol G, Delingette H, Garidel Y, Grevin G, Fidrich M, Bailet P and Ollier A. A Fully Three-dimensional Method for Facial Reconstruction based on Deformable Models, Journal of Forensic Science 1997 ; 42(4) : 649 :652.

[10] Bro-Nielsen M and **Cotin** S : Real-time volumetric deformable models for surgery simulation using finite elements and condensation, Eurographics'96, Computer Graphics Forum 1996 ; 15(3) :57-66.

Conference proceedings (peer reviewed)

[11] Dequidt J, Lenoir J, and **Cotin** S. Interactive Contacts Resolution Using Smooth Surface Deformation. Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), LNCS 4792, pp. 850-857, 2007.

[12] Wu X, Allard J, and **Cotin** S. Real-time Modeling of Vascular Flow for Angiography Simulation. Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), Volume 4792, pp. 850-857, 2007.

[13] Faure F, Allard J, **Cotin** S, Neumann P, Bensoussan P-J, Duriez C, Delingette H, Grisoni L. SOFA: a modular yet efficient simulation framework. Surgetica, 2007.

[14] Rabinov J, **Cotin** S, Allard J., J. Dequidt, J. Lenoir, V. Luboz , P. Neumann, X. Wu , S. Dawson. EVE: Computer Based Endovascular Training System for Neuroradiolgy. ASNR 45th Annual Meeting & NER Foundation Symposium, pp. 147-150, 2007.

[15] Allard J, **Cotin** S, Faure F, Bensoussan P-J, Poyer F, Duriez C, Delingette H, and Grisoni L. SOFA – an Open Source Framework for Medical Simulation. In Medicine Meets Virtual Reality (MMVR), 2007.

[16] Lenoir J, **Cotin** S, Duriez C, Neumann P. Physics-based Models for Catheter, Guidewire and Stent Simulation. Proceedings of 14th Annual Meeting, Medicine Meets Virtual Reality, Westwood et al eds ; 2006, p. 305-310

[17] Luboz V, Wu X, Krissian K, Westin C.F, Kikinis R, **Cotin** S, Dawson S. A segmentation and reconstruction technique for 3D vascular structures. Proceedings of the MICCAI Conference, MICCAI 2005, pp 43-50, Palm Spring, CA, October 2005.

[18] Cotin S, Duriez C, Lenoir J, Neumann P, Dawson S. New approaches to catheter navigation for interventional radiology simulation. Proceedings of the MICCAI Conference, MICCAI 2005, pp 534-542, Palm Spring, CA, October 2005.

[19] Cotin S, Neumann P, Wu X, Fonteneau, Bensoussan, P-J, Dequidt, J, Marchal D, Grisoni, L. Collaborative Development of an open framework for medical simulation. The Insight Journal, MICCAI Open-Source Workshop, 2005.

[20] Wu X, Krissian K, Luboz V, **Cotin** S, Dawson S. Smooth Vasculature Reconstruction from Patient Scan. Proceedings of the Virtual Environment Interactions and Physical Simulation workshop, VRIPHYS 2005, Pisa, Italy, November 2005.

[21] Lenoir J, **Cotin** S, Duriez C, Neumann P. Interactive physically-based simulation of Catheter and Guidewire. In proceedings of VRIPHYS'05 (Second Workshop in Virtual Reality Interactions and Physical Simulations), November 2005. Pisa, Italy. Best Paper Award.

[22] Valtorta D, Hollenstein M, Nava A, Luboz V, Lu M, Choi A, Mazza E, Zheng Y, **Cotin** S. Mechanical characterization of soft tissue: comparison of different experimental techniques on synthetic materials. Proceedings of the Fourth International Conference on the Ultrasonic Measurement and Imaging of Tissue Elasticity, p 94, Lake Travis, Austin, Texas, USA, October 16-19, 2005.

[23] Wu X, Pegoraro V, Luboz V, Neumann P, Bardsley R, Dawson S, **Cotin** S. New Approaches to Computer-based Interventional Neuroradiology Training. Proceedings of 13th Annual Meeting, Medicine Meets Virtual Reality; January 2005.

[24] P.F. Neumann, S. **Cotin**, X. Wu, V. Pegoraro, V. Luboz, and S. Dawson, High-Fidelity Interventional Neuroradiology Training System, Proceedings of the 30th Annual Society of Interventional Radiology Annual Scientic Meeting, April, 2005.

[25] Cotin S, Neumann P,Wu X, Pegoraro V, Bardsley R, Dawson S. Interventional Neuroradiology Training through Simulation. IR2 : Stroke Interventions, Society of Interventional Radiology ; January 2004.

[26] Stylopoulos N, **Cotin** S, Dawson S, Ottensmeyer M, Neumann P, Bardsley R, Russell M, Jackson P, Rattner D. CELTS: A clinically-based computer enhanced laparoscopic training system. Proceedings of 11th Annual Meeting, Medicine Meets Virtual Reality, pp. 336-342, 2003;

[27] Manivannan M, **Cotin** S, Srinivasan M, Dawson S. Real-Time PC based X-ray Simulation for Interventional Radiology Training. Proceedings of 11th Annual Meeting, Medicine Meets Virtual Reality, Westwood JD, Homan HM, Mogel GT Eds ; 2003, p. 233-239, 2003.

[28] Cotin S, Stylopoulos N, Ottensmeyer M, Neumann P, Rattner D, Dawson S. Metrics for Laparoscopic Skills Trainers : The Weakest Link ! Dohi T and Kikinis R, eds. Proceedings of MICCAI 2002, Lecture Notes in Computer Science 2488, p. 35-43, Springer-Verlag, 2002.

[29] Kerdok A, **Cotin** S, Ottensmeyer M, Galea A, Howe R, Dawson S. Truth Cube : Establishing Physical Standards for Soft Tissue Simulation. International Workshop on Deformable Modeling and Soft Tissue Simulation, Germany ; 2001 November 14-15.

[30] Cotin S, Shaffer D, Meglan D, Ottensmeyer M, Berry P, and Dawson S, CAML : a general framework for the development of medical simulation systems, Proceedings of the SPIE conference on Digitization of the Battlespace and Battleeld Biomedical Technologies, pp. 294-300, Orlando, 2000.

[31] Cotin S, Dawson S, Meglan D, Shaer D, Ferrell M, Bardsley R, Morgan F, Nagano T, Nikom J, Walterman M, Wendlandt J. ICTS, an Interventional Cardiology Training System. Medicine Meets Virtual Reality 2000, IOS Publishing, pp. 59-65, 2000.

[32] Delingette H, **Cotin** S and Ayache N, A Hybrid Elastic Model Allowing Real-Time Cutting, Deformations and Force-Feedback for Surgery Training and Simulation, Computer Animation 1999, May 1999, Geneva, Switzerland.

[33] Ayache N, **Cotin** S, and Delingette H. Surgery Simulation with visual and haptic feedback. In Y. Shirai and S. Hirose, editors, Robotics Research, the Eighth International Symposium, pp. 311-316. Springer, 1998.

[34] Cotin S, Delingette H, Clément JM, Bro-Nielsen M, Ayache N, and Marescaux J : Geometrical and Physical Representations for a Simulator of Hepatic Surgery, Proc. Medicine Meets Virtual Reality IV (MMVR' 96), 1996, pp. 139-150.

[35] Cotin S, Delingette H, and Ayache N. Real-time volumetric deformable models for surgery simulation. In Visualization in Biomedical Computing, Proceedings, volume 1131, Lecture Notes in Computer Science. Springer Verlag, New York, September 1996.

[36] Cotin S, Delingette H, Clément J-M, Tassetti V, Marescaux J, and Ayache N. Simulation de chirurgie hépatique avec système de retour de forces. In Interface to Real and Virtual Worlds, pp. 139-148, May 1996.

[37] Cotin S, Delingette H, and Ayache N. New perspectives for realistic simulations of hepatic surgery. In Actes du Congres Mondial de Télémédecine, Toulouse, France, 1995.

[38] Delingette H, Subsol G, **Cotin** S, and Pignon J. Simulation de chirurgie craniofaciale et realite virtuelle. In Interface des Mondes Reels et Virtuels (IMRV'94), Montpellier, pp. 399-408, January 1994.

Other publications

[39] Cotin S. Découpe de Surfaces à l'aide d'un Système de Réalité Virtuelle. Master's thesis, Université de Nice, Sophia-Antipolis, 1993.

[40] Delingette H, Subsol G, **Cotin** S, and Pignon J. A Craniofacial Surgery Testbed. Technical report 2199, I.N.R.I.A., Sophia-Antipolis, France, February 1994.

[41] Cotin S, Delingette H, and Ayache N. Efficient Linear Elastic Models of Soft Tissues for real-time surgery simulation. Rapport de Recherche 3510, INRIA, 1998.

[42] Cotin S, Delingette H, and Ayache N. Real-time elastic deformations of soft tissues for surgery simulation. Technical report RR-3511, INRIA, 1998.

Audio-visual material

N. Ayache, E. Bardinet, S. Benayoun, H. Delingette, G. Malandain, S. Fernández-Vidal, L. Soler, J. Montagnat, S. **Cotin**, G. Subsol, and J.-P. Thirion. “Analysis of medical images: Registration, Atlas, Motion”. Video, INRIA, 1996.

H. Delingette, J. Pignon, S. **Cotin**, and G. Subsol. “Cranio-facial Surgery Modeling”. Video, INRIA, October 1993. Note : Produced by INRIA media group.

BIBLIOGRAPHY

(Abrahamson, 1969)	S. Abrahamson, J. Denson, and R. Wolf. “Effectiveness of a simulator in training anesthesiology resident.” Journal of Medical Education, Vol. 44, 515-519, 1969.
(Acary, 2008)	V. Acary, B. Brogliato. “Numerical Methods for Non-smooth Dynamical Systems: Applications in Mechanics and Electronics”. Lecture Notes in Applied and Computational Mechanics, vol. 35. Springer Verlag, 2008.
(Adams, 1998)	R. Adams and B. Hannaford. "A Two-Port Framework for the Design of Unconditionally Stable Haptic Interfaces." In IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1254-1259, 1998.
(Alderliesten, 2004)	T. Alderliesten T. “Simulation of minimally-invasive vascular interventions for training purposes”. PhD dissertation, Utrecht University, December 2004.
(Allard, 2007)	J. Allard, S. Cotin, F. Faure, P.-J. Bensoussan, F. Poyer, C. Duriez, H. Delingette, and L. Grisoni. “SOFA – an Open Source Framework for Medical Simulation”. In Medicine Meets Virtual Reality (MMVR), 2007.
(Alterovitz, 2002)	R. Alterovitz, and K. Goldberg. “Comparing algorithms for soft tissue deformation: Accuracy metrics and benchmarks”. Technical report, UC Berkeley, 2002.
(Anderson, 2007)	A. Anderson, B. Ellis, and J. Weiss. “Verification, validation and sensitivity studies in computational biomechanics”. Computer Methods in Biomechanics and Biomedical Engineering 10(3): 171–184, 2007.
(Anitescu, 1999)	M. Anitescu, F. Potra, D. Stewart. “Time-stepping for three-dimentional rigid body dynamics”. Computer Methods in Applied Mechanics and Engineering (177),183–197, 1999.
(Arruda, 1993)	E. Arruda, and M. Boyce. “A Three-Dimensional Constitutive Model for the Large Stretch Behavior of Rubber Elastic Materials”. J. Mech. Phys. Solids; 41:389-412, 1993.
(Avila, 1996)	R. Avila and L. Sobierajski. “A haptic interaction method for volume visualization”. In Proceedings of Visualization ’96, pp 197–204, 1996.
(Ayache, 1998)	N. Ayache, S. Cotin, H. Delingette, J.-M. Clement, J. Marescaux, and M. Nord. “Simulation of Endoscopic Surgery”. Journal of Minimally Invasive Therapy and Allied Technologies; 7(2) :71-77, 1998.
(Bachofen, 2006)	D. Bachofen, J. Zátony, M. Harders, G. Székely, P. Frueh, and M. Thaler. “Enhancing the Visual Realism of Hysteroscopy Simulation”. Proc. of Medicine Meets Virtual Reality conference, pp. 31-36, 2006.
(Baciu, 2002)	G. Baciu, and W. Wong. “Hardware-assisted self-collision for deformable surfaces”. In Proceedings of ACM Symposium on Virtual Reality Software and Technology, pp. 129-136, 2002.

(Bacon, 2006)	J. Bacon, N. Tardella, J. Pratt, and J. English. “SSTML, the Surgical Simulation and Training Markup Language: An XML-Based Language for Medical Simulation”. In Proceedings of MMVR, pages 37–42, 2006.
(Balaniuk, 2003)	R. Balaniuk and K. Salisbury. “Soft-tissue simulation using the radial elements method”. Proceedings of Surgery Simulation and Soft Tissue Modeling, International Symposium (IS4TM), volume 2673 of Lecture Notes in Computer Science, pp 48–58, 2003.
(Baraff, 1998)	D. Baraff, A. Witkin. “Large steps in cloth simulation”. In Proc. SIGGRAPH ’98, pp. 43–54. ACM Press, 1998.
(Barea, 2007)	R. Barea, L. Boquete, J.F. Pérez, M. Dapena, P. Ramos and M. Hidalgo. “Cataract Surgery Simulator for Medical Education”. In proc. of the 11th Mediterranean Conference on Medical and Biomedical Engineering and Computing, pp. 1038-1042, 2007.
(Basdogan, 1998)	C. Basdogan, C. Ho, M. Srinivasan, S. Small, and S. Dawson. “Force Interaction in Laparoscopic Simulation: Haptic Rendering Soft Tissues”. In proceedings of Medecine Meets Virtual Reality (MMVR), pages 28-31, 1998.
(Basdogan, 2001)	C. Basdogan, C.-H. Ho and M. Srinivasan. “Virtual environments for medical training : Graphical and haptic simulation of laparoscopic common bile duct exploration”. In IEEE/ASME Transaction on mechatronics, Vol. 6, 2001.
(Bathe, 1996)	K. Bathe. “Finite Element Procedures”. Prentice Hall, 1996.
(Batteau, 2004)	L. Batteau, A. Liu, J. Maintz, Y Bhasin and M. Bowyer. “A Study on the Perception of Haptics in Surgical Simulation”. In Proc. International Symposium on Medical Simulation, pp. 185-192, 2004.
(Baudet, 2007)	V. Baudet, M. Beuve, F. Jaillet, B. Shariat, F. Zara. “Integrating tensile parameters in 3D mass-spring system”. In: Proceedings of Surgetica, 2007.
(Berkley, 1999)	J. Berkley, S. Weghorst, H. Gladstone, G. Raugi, D. Berg, and M. Ganter. “Banded matrix approach to finite element modeling for soft tissue simulation.”, Virtual Reality, 4(3):203–212, 1999.
(Bianchi, 2004)	G. Bianchi, B. Solenthaler, G. Szkely, and M. Harders. “Simultaneous topology and stiffness identification for mass-spring models based on FEM reference deformations”. In: Proceedings of MICCAI, pp. 293–301, 2004.
(Bielser, 1999)	D. Bielser, V. Maiwald, and M. Gross. “Interactive cuts through 3-dimensional soft tissue”. In Computer Graphics Forum (Eurographics ’99), volume 18, pp 31–38, 1999.
(Bielser, 2000)	D. Bielser and M. Gross. “Interactive simulation of surgical cuts”. In Proceedings of the Eighth Pacific Conference on Computer Graphics and Applications, pp. 116–125, 2000.
(Biermann, 2000)	H. Biermann, A. Levin, and D. Zorin. “Piecewise smooth subdivision surfaces with normal control”. Proc. of SIGGRAPH conference, pp. 113-20, 2000.
(Bland, 1986)	J. Bland, and D. Altman. “Statistical methods for assessing agreement between two methods of clinical measurement”. Lancet, pp. 307-310, 1986.

(Bowyer, 2006)	M. Bowyer. “Validation of simulation: comparison to porcine models”. Annual TATRC Advanced Medical Technology Review, held at the 14th Annual Medicine Meets Virtual Reality (MMVR) conference, 2006.
(Bro-Nielsen, 1996)	M. Bro-Nielsen and S. Cotin. “Real-time Volumetric Deformable Models for Surgery Simulation using Finite Elements and Condensation”. In Eurographics Computer Graphics Forum 15 (3), 57–66, 1996.
(Bühler, 2002)	K. Bühler, P. Felkel, and A. La Cruz. “Geometric Methods for Vessel Visualization and Quantification - A Survey”. VRVis Research Center, Vienna, Austria, Tech Report, pp. 24-48, 2002.
(Chabanas, 2004)	M. Chabanas, Y. Payan, C. Marcaux, P. Swider, F. Boutault. “Comparison of linear and non-linear soft tissue models with post-operative CT scan in maxillofacial surgery. In: International Symposium on Medical Simulation, pp. 19–27, 2004.
(Clatz, 2004)	O. Clatz, P.-Y. Bondiau, H. Delingette, G. Malandain, M. Sermesant, S. Warfield, and N. Ayache. “In Silico Tumor Growth: Application to Glioblastomas”. Proc. of the conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer Verlag LNCS, volume 3217, pp. 337-345, 2004.
(Colgate, 1995)	J. Colgate, C. Stanley, and J. Brown. "Issues in the Haptic Display of Tool Use." In Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 140-145, 1995.
(Comas, 2008)	O. Comas, Z. Taylor , J. Allard, S. Ourselin, S. Cotin, and J. Passenger. “Efficient nonlinear FEM for soft tissue modeling and its GPU implementation within the open source framework SOFA”. In Proc. International Symposium on Computational Models for Biomedical Simulation - ISBMS, 2008 (to appear).
(Costa, 2000)	I. Costa and R. Balaniuk. “LEM - An approach for physically based soft tissue simulation suitable for haptic interaction. In Proceedings of the Fifth PHANTOM Users Group Workshop, 2000.
(Cotin, 1996a)	S. Cotin, H. Delingette, J.-M. Clément, M. Bro-Nielsen, N. Ayache, and J. Marescaux. “Geometrical and Physical Representations for a Simulator of Hepatic Surgery”, Proc. Medicine Meets Virtual Reality IV (MMVR' 96), 1996, pp. 139-150.
(Cotin, 1996b)	S. Cotin, H. Delingette, and N. Ayache. “Real-time volumetric deformable models for surgery simulation”. In Visualization in Biomedical Computing, Proceedings, volume 1131, Lecture Notes in Computer Science, 1996.
(Cotin, 1999)	S. Cotin, H. Delingette, and N. Ayache. “Real-time elastic deformations of soft tissues for surgery simulation”. IEEE Transactions On Visualization and Computer Graphics 5(1): 62-73, 1999.
(Cotin, 2000a)	S. Cotin, H. Delingette, and N. Ayache. “A hybrid elastic model for real-time cutting, deformations, and force feedback for surgery training and simulation”. The Visual Computer, 16(8):437-452, 2000.

(Cotin, 2000b)	S. Cotin, D. Shaffer, D. Meglan, M. Ottensmeyer, P. Berry, and S. Dawson. “CAML: a general framework for the development of medical simulation systems”. Proceedings of the SPIE conference on Digitization of the Battlespace and Battleeld Biomedical Technologies, pp. 294-300, 2000.
(Cotin, 2000c)	S. Cotin, S. Dawson, D. Meglan, D. Shaffer, M. Ferrell, R. Bardsley, F. Morgan, T. Nagano, J. Nikom, M. Walterman, and J. Wendlandt. “ICTS, an Interventional Cardiology Training System”. Medicine Meets Virtual Reality 2000, IOS Publishing, pp. 59-65, 2000.
(Cotin, 2002)	S. Cotin, N. Stylopoulos, M. Ottensmeyer, P. Neumann, D. Rattner, and S. Dawson. “Metrics for Laparoscopic Skills Trainers : The Weakest Link!”. In Proceedings of MICCAI 2002, Lecture Notes in Computer Science 2488, p. 35-43, Springer-Verlag, 2002.
(Cotin, 2005)	S. Cotin, C. Duriez, J. Lenoir, P. Neumann, and S. Dawson. New approaches to catheter navigation for interventional radiology simulation. Proceedings of the MICCAI Conference, pp. 534-542, 2005.
(Cover, 1993)	S. Cover, N. Ezquerra, J. O’Brien, R. Rowe, T. Gadacz, and E. Palm. “Interactively deformable models for surgery simulation”. IEEE Computer Graphics and Applications, 13(6):68–75, 1993.
(De, 2001)	S. De, J. Kim, and M. Srinivasan. “A Meshless Numerical Technique for Physically Based Real Time Medical Simulations”. In proceedings of Medicine Meets Virtual Reality (MMVR), IOS Press, pp. 113-118, 2001.
(Dawson, 2000)	S. Dawson, S. Cotin, D. Meglan, D. Shaffer, and M. Ferrell, “Designing a computer-based simulator for interventional cardiology training”, Catheterization and Cardiovascular Interventions, vol. 51, 2000, pp. 522-527.
(Debunne, 2002)	G. Debunne, M. Desbrun, A. Barr, M.-P. Cani. “Interactive multi-resolution animation of deformable models”. In Eurographics Workshop on Computer Animation and Simulation ’99 (1999), pp. 133–144. 5for the simulation of deformable models. The Visual Computer (2002).
(Delingette, 1997)	H. Delingette. “General object reconstruction based on simplex meshes”. Technical Report 3111, INRIA, February 1997.
(Delingette, 1999)	H. Delingette, S. Cotin, and N. Ayache. “A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation”. In Proceedings of Conference on Computer Animation, 1999.
(Dequidt, 2007)	J. Dequidt, J. Lenoir, and S. Cotin. “Interactive Contacts Resolution Using Smooth Surface Deformation”. Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), LNCS 4792, pp. 850-857, 2007.
(Dequidt, 2008)	J. Dequidt, M. Marchal, C. Duriez, and S. Cotin. “Interactive Simulation of Embolization Coils: Modeling and Experimental Validation”. Submitted to the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), 2008.

(DiMaio, 2003)	S. DiMaio and S. Salcudean. “Needle Insertion Modeling and Simulation”. In IEEE Transactions on Robotics and Automation, 19(5): 864- 875, 2003.
(Duriez, 2006a)	C. Duriez , S. Cotin , J. Lenoir , and P. Neumann, “New Approaches to Catheter Navigation for Interventional Radiology Simulation”, Computer Aided Surgery, vol. 11(6), 2006, pp. 300-308.
(Duriez, 2006b)	C. Duriez, F. Dubois, A. Kheddar, and C. Andriot. “Realistic haptic rendering of interacting deformable objects in virtual environments”. IEEE Transactions on Visualization and Computer Graphics, 12(1):36–47, 2006.
(Eason, 2005)	M. Eason, D. Linville, and C. Stanton. “A System to Simulate Arterial Blood Flow for Cannulation in the Human Patient Simulator.” Anesthesiology; 103(2):443, 2005.
(Faure, 2007)	F. Faure, J. Allard, P. Neumann, P.-J. Bensoussan, C. Duriez, H. Delingette and L. Grisoni. “SOFA - a Modular Yet Efficient Simulation Framework”. Proc. Surgetica, 2007.
(Featherstone, 1983)	R. Featherstone, "The Calculation of Robot Dynamics using Articulated-Body Inertias", Int. J. Robotics Research, vol. 2, no. 1, pp. 13-30, 1983.
(Felippa, 2000)	C. Felippa. “A systematic approach to the element independent corotational dynamics of finite elements”. Technical Report CU-CAS-00-03, Center for Aerospace Structures, 2000.
(Felkel, 2004)	P. Felkel, R. Wegenkittl, and K. Bühler. “Surface Models of Tube Trees”. In: Computer Graphics International (CGI'04), pp. 70-77, 2004.
(Ferrant, 2001)	M. Ferrant, A. Nabavi, B. Macq, F. Jolesz, R. Kikinis, S. Warfield. Registration of 3-D intraoperative MR images of the brain using a finite element biomechanical model”. IEEE Trans Med Imaging; 20(12):1384-1397, 2001.
(FitzHugh, 1961)	R.A. FitzHugh. “Impulses and physiological states in theoretical models of nerve membrane”. In Biophysical Journal, vol. 1, pp. 445-466, 1961.
(Fitzgibbon, 1999)	A. Fitzgibbon, M. Pilu, and R. Fisher. “Direct least-squares fitting of ellipses”. IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(5):476-480, 1999.
(Fung, 1993)	Y.C. Fung: “Biomechanics: Mechanical Properties of Living Tissues”. Second edition, Springer–Verlag, 1993.
(Gibson, 1998)	S. Gibson, J. Samosky, A. Mor, C Fyock, E. Grimson, T. Kanade, R. Kikinis, H. Lauer, N. McKenzie, S. Nakajima, H. Ohkami, R. Osborne, and A. Sawada. “Simulating Surgery Using Volumetric Object Representations, Real-time Volume Rendering, and Haptic Feedback”, Medical Image Analysis, Vol. 2, No. 2, 1998, pp. 121-132.
(Golub, 1996)	G. Golub and C. Van Loan. “Matrix Computations (third ed.)”.Johns Hopkins University Press, Baltimore, USA, 1996.
(Gordon, 2006)	J. Gordon, D. Brown, and E. Armstrong. “Can a simulated critical care encounter accelerate basic science learning among preclinical medical students? A pilot study”. Journal of the Society for Medical Simulation, Vol. 1, pp. 13-17, 2006.

(Goktekin, 2004)	T. Goktekin, M. Cenk Cavusoglu, and F. Tendick. “GiPSi: An open source software development framework for surgical simulation”. In International Symposium on Medical Simulation, pp. 240–248, 2004.
(Govindaraju, 2006)	N. Govindaraju, I. Kabul, M. Lin, D. Manocha. “Efficient Continuous Collision Detection in Deformable Models using GPUs”. Elsevier Computers & Graphics, Special issue on Eurographics Symposium on Virtual Environments, 2006.
(Guyton, 1995)	A. Guyton, and J. Hall. Textbook of Medical Physiology,ISBN 0721659446, 1995.
(Hauser, 2003)	K. Hauser, C. Shen and J. O'Brien. “Interactive deformation using modal analysis with constraints”. Proceedings of Graphics Interface, pp. 247-255, 2003.
(Hauth, 2003)	M. Hauth, O. Etzmuß, and W. Straßer. “Analysis of numerical methods for the simulation of deformable models”. The Visual Computer 19(7-8): 581-600, 2003.
(Heidelgerg, 2003)	B. Heidelgerg, M. Teschner, M. Gross. “Real-time volumetric intersections of deforming objects”. In Proc. of Vision, Modeling, Visualization VMV’03, pp. 461-468, 2003.
(Heiland, 2004)	M. Heiland, A. Petersik, B. Pflesser, U. Tiede, R. Schmelzle, K.-H. Höhne, and H. Handels. “In Proc. Computer Assisted Radiology and Surgery, Volume 1268, pp. 1226-1229, 2004.
(Hirota, 1999)	G. Hirota, R. Maheshwari, and M.C. Lin. “Fast volume preserving free form deformation using multilevel optimization”. In SMA ’99: Proceedings of the fifth ACM symposium on Solid modeling and applications, pp. 234–245, 1999.
(Hoefer, 2002)	U. Hoefer, T. Langen, J. Nziki, O. Schmid, F. Zeitler, J. Hesser, W. Voelker, and R. Maenner, “Cath-I - Catheter Instruction System”, in Proc. of Computer Assisted Radiology and Surgery (CARS), 2002.
(Hollenstein, 2006)	M. Hollenstein, A. Nava, D. Valtorta, J. G. Snedeker and E. Mazza. “Mechanical Characterization of the Liver Capsule and Parenchyma”. Proc. MICCAI conference. Lecture Notes in Computer Science, 4072, pp. 150-158, (2006).
(Holzapfel, 2000)	G.A. Holzapfel and C.A.J. Schulz-Bauer and M. Stadler. “ Mechanics of Angioplasty: Wall, Balloon, and Stent”. Mechanics in Biology, 142:141-156 , 2000.
(Hoppe, 1993)	H. Hoppe, T. DeRose, T. DuChamp, J. McDonald, and W. Steutze. “Mesh optimization”. In Proceedings of SIGGRAPH ’93, pages 19–26, 1993.
(Howell, 1998)	D. Howell (ed.). “Fundamental Statistics for the Behavioral Sciences”. Duxbury Press Pacific Grove, CA, 1998.
(Hu, 2001)	S. Hu, H. Zhang, C. Tai, and J. Sun. “Direct manipulation of FFD: effcient explicit solutions and decomposible multiple point constraints”. Visual Computer, 17(6): 370–379, 2001.
(Hu, 2006)	J. Hu, C. Chang, N. Tardella, J. Pratt, and J. English. “Effectiveness of haptic feedback in open surgery simulation and training systems”. Proc. Medicine Meets Virtual Reality, pp. 213-218, 2006.
(Humphrey, 2002)	J. Humphrey. “Continuum Biomechanics of Soft Biological Tissues”. Proc. R. Soc. Lond. pp. 1-43, 2002.

(Immersion)	Immersion Medical web site, www.immersion.com
(Irving, 2004)	G. Irving, J. Teran and R. Fedkiw. “Invertible finite elements for robust simulation of large deformation”. Proceedings of Eurographics Symposium on Computer Animation, 2004.
(Jambon, 1998)	A.-C. Jambon, F. Dubecq-Princeteau, P. Dubois, S. Karpf, C. Chaillou, P. Meseure, and D. Querleu. “SPIC: a training simulator for coelioscopic interventions with gynecologic purpose”. J Gynecol Obstet Biol Reprod, 27 (5):536-43, 1998.
(James, 1999)	D. James and D. Pai. “Artdefo: Accurate real time deformable objects”. Proceedings of SIGGRAPH, pp. 65-72, 1999.
(Kass, 1987)	M. Kass, A. Witkin, and D. Terzopoulos. “Snakes: Active Contour Models”. In Proceedings of the first International Conference on Computer Vision, pp. 259-267, 1987.
(Kawasaki, 2005)	M. Kawasaki, M. Rissanen, N. Kume, Y. Kuroda, M. Nakao, T. Kuroda, and H. Yoshihara. VRASS (Virtual Reality Aided Simulation, www.kuhp.kyoto-u.ac.jp/~mvl/)
(Keeve, 1999)	E. Keeve, S. Girog, R. Kikinis, and B. Girod. “Deformable modeling of facial tissue for craniofacial surgery simulation”. Computer Aided Surgery, 3(5):228–238, 1999.
(Kerdok, 2003)	A. Kerdok, S. Cotin, M. Ottensmeyer, A. Galea, R. Howe, and S. Dawson. “Truth Cube: Establishing Physical Standards for Real Time Soft Tissue Simulation”. Medical Image Analysis; 7:283-291, 2003.
(Kerdok, 2005)	A. Kerdok, M. Ottensmeyer, and R. Howe. “Effects of perfusion on the viscoelastic characteristics of liver”. J Biomech, 39(12):2221-2231, 2005.
(Kerdok, 2006)	Amy Kerdok. "Characterizing the Nonlinear Mechanical Response of Liver to Surgical Manipulation". Ph.D thesis, Harvard University, 2006.
(Kohn, 2000)	L. Kohn, J. Corrigan, and M. Donaldson. “To Err is Human: Building a Safer Health System”. Editors. Committee on Quality of Health Care in America, Institute of Medicine, National Academy Press, 2000.
(Krissian, 2002)	K. Krissian. “Flux-based anisotropic diffusion applied to enhancement of 3-D angiogram”. IEEE Transactions in Medical Imaging, 21:1440-52, 2002.
(Krissian, 2005)	K. Krissian, and C.-F. Westin. “Fast Sub-Voxel Re-initialization of the Distance Map for Level Set Methods”, Pattern Recognition Letters, 26:10, pp. 1532-1542, 2005.
(Kühnapfel, 1995)	U. Kühnapfel, H. Krumm, C. Kuhn, M. Hübner, and B. Neisius. “Endosurgery simulations with KISMET: a flexible tool for surgical instrument design, operation room planning and VR technology based abdominal surgery training”. In Proc. of the Virtual Reality World conference, pp. 165-171, 1995.
(Kühnapfel, 2000)	U. Kühnapfel, H. Çakmak, and H. Maass. “Endoscopic Surgery Training using Virtual Reality and deformable Tissue Simulation”. Computers & Graphics 24(5): 671-682, 2000.
(Kumar, 1993)	S. Kumar, and A. Petho. “An algorithm for the numerical inversion of a tri-diagonal matrix”. Communications in Numerical Methods in Engineering 9(4) 353-359, 1993.

(Lagarias, 1998)	J. Lagarias, J. Reeds, M. Wright, and P. Wright. “Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions”. SIAM Journal of Optimization; 9:112-147, 1998.
(Lenoir, 2002)	J. Lenoir, P. Meseure, L. Grisoni, and C. Chaillou. “Surgical thread simulation”. In Proc. MS4CMS (Modelling & Simulation for Computer Aided Medicine and Surgery). European Series in Applied and Industrial Mathematics, Vol 12, pp 102-107, 2002.
(Lenoir, 2004)	J. Lenoir, and S. Fonteneau. “Mixing deformable and rigid-body mechanics simulation”. In: Computer Graphics International, pp. 327–334, 2004.
(Lenoir, 2005)	J. Lenoir, S. Cotin, C. Duriez, and P. Neumann. “Interactive physically-based simulation of Catheter and Guidewire”. In proceedings of VRIPHYS'05 (Second Workshop in Virtual Reality Interactions and Physical Simulations), 2005.
(Lenoir, 2006)	J. Lenoir, S. Cotin, C. Duriez, P. Neumann. “Physics-based Models for Catheter, Guidewire and Stent Simulation”. Proceedings of 14th Annual Meeting, Medicine Meets Virtual Reality, pp. 305-310, 2006.
(Leskowsky, 2006)	R. Leskowsky, M. Cooke, M. Ernst, and M. Harders. “Using multidimensional scaling to quantify the fidelity of haptic rendering of deformable objects”. In Proceedings of EuroHaptics, pp. 289–295, 2006.
(Lin, 2008)	M. Lin, M. Otaduy, and A. Peters. “Haptic Rendering”, 2008.
(Lloyd, 2007)	B. Lloyd, D. Szczerba, and G. Székely. “A Coupled Finite Element Model of Tumor Growth and Vascularization”. In Proc. of the conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), Lecture Notes in Computer Science, volume 4792, pp. 874-881, 2007.
(Lorensen, 1987)	W. Lorensen and H. Cline. “Marching Cubes: A high resolution 3-D surface construction algorithm”. Computer Graphics, vol. 21, pp. 163-169, 1987.
(Luboz, 2005)	V. Luboz, X. Wu, K. Krissian, C.-F. Westin, R. Kikinis, S. Cotin, and S. Dawson. “A segmentation and reconstruction technique for 3D vascular structures”. Proceedings of the MICCAI Conference, pp. 43-50, 2005.
(Maithel, 2006)	S. Maithel, R. Sierra, J. Korndorffer, P. Neumann, S. Dawson, M. Callery, D. Jones, and D. Scott. “Construct and face validity of MIST-VR, Endotower, and CELTS: are we ready for skills assessment using simulators?”. Surgical Endoscopy, 20(1): 104-112, 2006.
(Manivannan, 2003)	M. Manivannan, S. Cotin, M. Srinivasan, and S. Dawson. “Real-Time PC based X-ray Simulation for Interventional Radiology Training”. Proceedings of Medicine Meets Virtual Reality, pp. 233-239, 2003.
(Marchal, 2008)	M. Marchal, J. Allard, C. Duriez, and S. Cotin. “Towards a Framework for Assessing Deformable Models in Medical Simulation”. In Proc. International Symposium on Computational Models for Biomedical Simulation - ISBMS, 2008 (to appear).

(Marchal, 2006)	M. Marchal, E. Promayon, and J. Troccaz. “Simulating Prostate Surgical Procedures with a Discrete Soft Tissue Model”. Third Eurographics Workshop in Virtual Reality, Interactions, and Physical Simulations (VriPhys'06), pp. 109-118, 2006.
(Marescaux, 1998)	J. Marescaux, J.M. Clement, V. Tassetti, C. Koehl, S. Cotin, Y. Russier, D. Mutter, H. Delingette, and N. Ayache. “Virtual reality applied to hepatic surgery simulation : The next revolution”. Annals of Surgery 1998, 228(5) : 627-634.
(Masuzawa, 1992)	T. Masuzawa, Y. Fukui, and N. Smith. Cardiovascular simulation using a multiple modeling method on a digital computer”. Journal of Clinical Monitoring and Computing, 8(1):50-58, 1992.
(Mavrogiorgou, 2001)	P. Mavrogiorgou, R. Mergl, *et al.* “Kinematic analysis of handwriting movements in patients with obsessive-compulsive disorder”. J. Neurol. Neurosurg. Psychiatry 70(5):605-612, 2001.
(McAllister, 1975)	R. McAllister, D. Noble, and R. Tsien. “Reconstruction of the Electrical Activity of Cardiac Purkinje Fibres”, Journal of Physiology, vol. 251, pp. 1-59, 1975.
(McNeely, 1999)	W. McNeely, K. Puterbaugh, and J. Troy. "Six Degree-of-Freedom Haptic Rendering using Voxel Sampling." In Proceedings of SIGGRAPH '99, Computer Graphics Proceedings, Annual Conference Series, pp. 401-408, 1999.
(Mentice)	Mentice web site, www.mentice.com/
(Meseure, 1995)	P. Meseure, J-F. Rouland, P. Dubois, S. Karpf, and C. Chaillou. “SOPHOCLE: a retinal laser photocoagulation simulator overview”. In Proc. of the first conference in Computer Vision, Virtual Reality and Robotics in Medicine (CVRMed), Lecture Notes in Computer Science, vol. 905, pp. 103-114, 1995.
(Meseure, 2003)	P. Meseure, J. Davanne, L. Hilde, J. Lenoir, L. France, F. Triquet, C. Chaillou. “A Physically-Based Virtual Environment dedicated to Surgical Simulation”. In Proc. of Surgery Simulation and Soft Tissue Modeling (IS4TM), pp. 38-47, 2003.
(Miller, 2005)	K. Miller, Z. Taylor, and W. Nowinski. "Towards computing brain deformations for diagnosis, prognosis and neurosurgical simulation", J. Mechanics in Medicine and Biology, 5(1), pp.105-121, 2005.
(Montagnat, 1997)	J. Montagnat, H. Delingette, N. Ayache, J.M. Clément, C. Roy, Y. Russier, V. Tassetti, and J. Marescaux. “Liver segmentation in contrast enhanced helical CT scans”. In Proceedings of World Congress on Medical Physics and Biomedical Engineering, Nice, France, 1997.
(Montgomery, 2002)	K. Montgomery, C. Bruyns, J. Brown, S. Sorkin, F. Mazzella, G. Thonier, A. Tellier, B. Lerman, and A. Menon. “SPRING: A general framework for collaborative, real-time surgical simulation”. In Proceedings of MMVR, pages 23–26, 2002.
(Moreau, 1996)	J.-J. Moreau, and M. Jean. “Numerical treatment of contact and friction: the contact dynamics method. Engineering Systems Design and Analysis 4, 201–208, 1996.
(Mosegaard, 2004)	J. Mosegaard. “Parameter optimization for the behavior of elastic models over time”. Proceedings of Medicine Meets Virtual Reality 12, pp 253–255, 2004.

(Mosegaard, 2005)	J. Mosegaard, and T. Sorensen. “GPU accelerated surgical simulators for complex morphology”. IEEE Virtual Reality, 147-153, 2005.
(Muellller, 2004)	M. Mueller and M. Gross. “Interactive virtual materials”. Proceedings of the conference on Graphics interface. pp. 239 - 246, 2004.
(Mullick, 1996)	R. Mullick, H. T. Nguyen, Y. P. Wang, J. K. Raphel, and R. Raghavan. “Overview of Visible Human based applications at CieMed”, The Visible Human Project Conference, 1996.
(Napel, 1991)	S. Napel, S. dunne, and B. Rutt. “Fast Fourier Projection for MR angiography”. Magnetic Resonance in Medicine, volume 19, pp. 393-405, 1991.
(Nesme, 2005)	Nesme, Y. Payan and F. Faure. “Efficient, Physically Plausible Finite Elements”. Proceedings of the Eurographics conference, 2005.
(Neumann, 1998)	P. Neumann, L. Sadler, and J. Gieser. “Virtual Reality Vitrectomy Simulator”. In Proc. of the Medical Image Computing and Computer-Assisted Interventions conference (MICCAI), pp. 910-917, 1998.
(Neumann, 2005)	P. Neumann, S. Cotin, X. Wu, V. Pegoraro, V. Luboz, and S. Dawson. “High-Fidelity Interventional Neuroradiology Training System”. In Proceedings of the 30th Annual Society of Interventional Radiology Annual Scientific Meeting, 2005.
(Nienhuys, 2001)	H. Nienhuys and A. van der Stappen. “Supporting cuts and finite element deformation in interactive surgery simulation”. Technical Report UU-CS-2001-16, Institute of Information and Computing Sciences, Utrecht Universtiy, NL, June 2001.
(Noble, 1962)	D. Noble. “A Modification of the Hodgkin-Huxley Equations Applicable to Purkinje Fibre Action and Pace-maker Potentials”. In Journal of Physiology, vol. 160, pp. 317-352, 1962.
(Nowinski, 2001)	W.L. Nowinski, and C-K. Chui. “Simulation of interventional neuroradiology procedures”, in Proc. of International Workshop on Medical Imaging and Augmented Reality, 2001, pp. 87-94.
(O’Brien, 1999)	J. O’Brien, and K. Hodgins. “Graphical modeling and animation of brittle fracture”. In Proceedings of SIGGRAPH, pp. 287–296, 1999.
(Osher, 1988)	S. Osher, and J.A. Sethian. “Fronts propagating with curvature dependent speed: algorithms based on the Hamilton-Jacobi formalism”. J. Comput. Physics, 79 , 12-49, 1988.
(Otaduy, 2006)	M. Otaduy and M. Lin. "A Modular Haptic Rendering Algorithm for Stable and Transparent 6-DOF Manipulation." IEEE Transactions on Robotics 22(4):751-762, 2006.
(Ottensmeyer, 2001)	M. Ottensmeyer, and K. Salisbury. “In Vivo Data Acquisition Instrument for Solid Organ Mechanical Property Measurement”. Proceedings of Medical Image Computing and Computer-Assisted Intervention conference (MICCAI); Lecture Notes in Computer Science, Volume 2208, pp. 975-982, 2001.

(Park, 1996)	Y. Park, R. Lindeman, and J. Hahn. “X-ray Casting: Fast Volume Visualization Using 2D Texture Mapping Techniques”. Seventh IEEE Visualization conference, San Francisco, 1996.
(Pham, 2000)	D. Pham, C. Xu, and J. Prince. “Current Methods in Medical Image Segmentation”. Annual Review of Biomedical Engineering, pp. 315-337, 2000.
(Picinbono, 2001)	G. Picinbono, H. Delingette, and N. Ayache. “Non-linear and anisotropic elastic soft tissue models for medical simulation”. In proceedings of the IEEE International Conference Robotics and Automation (ICRA), pp. 1370 - 1375, 2001.
(Pidaparti, 2003)	R. Pidaparti, P. Sarma, and R. Meiss. “Smooth Muscle Tissue Mechanical Properties - Material Models”. Proceedings of the Biomechanics conference, ISBN: 0-88986-359-8, pp. 387-398, 2003.
(Platt, 1988)	J. Platt and A. Barr. “Constraint methods for flexible models”. In Proc. SIGGRAPH, pages 279–288, 1988.
(Provot, 1995)	X. Provot. “Deformation constraints in a mass-spring model to describe rigid cloth behavior”. In Proceedings of Graphics Interface, pages 141–155, 1995.
(Przemieniecki, 1968)	J.S. Przemieniecki. Theory of Matrix Structural Analysis. 1968.
(Rabinov, 2007)	J. Rabinov, S. Cotin, J. Allard, J. Dequidt, J. Lenoir, V. Luboz, P. Neumann, X. Wu, and S. Dawson. “EVE: Computer Based Endovascular Training System for Neuroradiolgy”. ASNR 45th Annual Meeting & NER Foundation Symposium, pp. 147-150, 2007.
(Radetzky, 2000)	A. Radetzky, A. Nurnberger, and D. Pretchner. “Elastodynamic shape modeler: A tool for defining the deformation behaviour of virtual tissues”. Radiographics, 20:865–881, 2000.
(Raghupathi, 2004)	L. Raghupathi, L. Grisoni, F. Faure, D. Marchal, M.-P. Cani, and C. Chaillou. “An intestinal surgery simulator: real-time collision processing and visualization”. IEEE Trans Vis Comput Graph. ;10 (6):708-18, 2004.
(Redon, 2002)	S. Redon, A. Khedday, and S. Coquillart. “Fast continuous collision detection between rigid bodies”. Computer Graphics Forum (Eurographics) 21(3): 279-288, 2002.
(Rosser, 1998)	J. Rosser, L. Rosser, and R. Savalgi. “Objective evaluation of a laparoscopic surgical skill program for residents and senior surgeons”. Arch. Surg. 133(6): 657-661, 1998.
(Ruspini, 1997)	D. Ruspini, K. Kolarov, and O. Khatib. "The Haptic Display of Complex Graphical Environments." In Proceedings of SIGGRAPH 97, Computer Graphics Proceedings, Annual Conference Series, pp. 345–352, 1997
(Santerre, 2006)	N. Santerre, F. Blondel, F. Racoussot, G. Laverdure, S. Karpf, P. Dubois, and J-F. Rouland. “A teaching medical simulator: phacoemulsification in virtual reality”, J Fr Ophtalmol.; 30 (6):621-626, 2006.
(Saupin, 2008a)	G. Saupin, C. Duriez, L. Grisoni, S. Cotin. “Efficient Contact Modeling using Compliance Warping”. In Proc. Computer Graphics International, 2008 (to appear).

(Saupin, 2008b)	G. Saupin, C. Duriez, and S. Cotin. “Contact model for haptic medical simulations”. In Proc. International Symposium on Computational Models for Biomedical Simulation - ISBMS, 2008 (to appear).
(Sederberg, 1986)	T. Sederberg and S. Parry. “Free-form deformation of solid geometric models”. Computer Graphics, 20(4):151–160, Aug 1986.
(Sermesant, 2006)	M. Sermesant, H. Delingette, and N. Ayache. ”An Electromechanical Model of the Heart for Image Analysis and Simulation”. In IEEE Transactions in Medical Imaging, 25(5): 612-625, 2006.
(Sethian, 1999)	J. Sethian, 1999. “Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational, Geometry, Fluid Mechanics, Computer Vision and Materials Science”, Cambridge Univ. Press
(Seymour, 2002)	N. Seymour, A. Gallagher, S. Roman, M. O’Brien, V. Bansal, D. Andersen and R. Satava. “Virtual Reality Training Improves Operating Room Performance: Results of a Randomized, Double-blinded Study”. Ann Surg. 2002; 236(4):458-63.
(Shaffer, 1999)	D. Shaffer, D. Meglan, M. Ferrell, and S. Dawson. “Virtual Rounds: simulation-based education in procedural medicine”. In Proceedings of SPIE Vol. 3712: Battlefield Biomedical Technologies, 1999.
(Sickle, 2005)	K. Sickle, D. McClusky III, A. Gallagher, snd C. Smith. “Construct validation of the ProMIS simulator using a novel laparoscopic suturing task”. Surgical Endoscopy. 2005, Vol. 19, No. 9: 1227.
(Simbionix)	Simbionix web site, www.simbionix.com
(SimSurgery)	Sim Surgery web site, www.simsurgery.com
(Sorensen, 2006)	T.S. Sorensen and J. Mosegaard. “Haptic feedback for the GPU-based surgical simulator”. In Proc. of Medicine Meets Virtual Reality, pp. 523-528, 2006.
(Stylopoulos, 2003)	N. Stylopoulos, S. Cotin, S. Dawson, M. Ottensmeyer, P. Neumann, R. Bardsley, M. Russell, P. Jackson, and D. Rattner. “CELTS: A clinically-based computer enhanced laparoscopic training system:. Proceedings of 11th Annual Meeting, Medicine Meets Virtual Reality, pp. 336-342, 2003.
(Stylopoulos, 2004)	N. Stylopoulos, S. Cotin, S. Maithel, M. Ottensmeyer, P. Jackson, Bardsley, P. Neumann, D. Rattner, and S. Dawson. “Computer-enhanced laparoscopic training system (CELTS): bridging the gap”. Surg Endosc. 18 (5):782-9, 2004.
(Sundaramoorthi, 2006)	G. Sundaramoorthi, A. Yezzi, and A. Mennucci. “Tracking with Sobolev active contours”. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Volume 1, pp. 674--680, 2006.
(Suzuki, 2001)	N. Suzuki, A. Hattori, S. Suzuki, M. Baur, A. Hirner, S. Kobayashi, Y. Yamazaki, and Y. Adachi. “Real-Time Surgical Simulation with Haptic Sensation as Collaborated Works between Japan and Germany”. Proceeding of MICCAI 2001, pp. 1015-1021, 2001.

(Szekely, 2000)	G. Szekely, C. Brechbuhler, J. Dual, R. Enzler, J. Hug, R. Hutter, N. Ironmonger, M. Kauer, V. Meier, P. Niederer, A. Rhomberg, P. Schmid, G. Schweitzer, M. Thaler, V. Vuskovic, G. Troster, U. Hailer, and M. Bajka. “Virtual reality-based simulation of endoscopic surgery”. Presence: Teleoperators & Virtual Environments, 9(3):310–324, 2000.
(Terzopoulos, 1987)	D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. “Elastically deformable models. In Computer Graphics Proceedings”, Proc. SIGGRAPH, pages 205–214, 1987.
(Terzopoulos, 1991)	D. Terzopoulos and D. Metaxas. “Dynamic 3d models with local and global deformations: deformable superquadrics”. In IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 13, pages 703–714, July 1991.
(Teschner, 2005)	M. Teschner, S. Kimmerle, G. Zachmann, B. Heidelberger, L. Raghupathi, A. Fuhrmann, M-P. Cani, F. Faure, N. Magnetat-Thalmann, W. Strasser. “Collision Detection for Deformable Objects”. Computer Graphics Forum, vol. 24, no. 1, pp. 61-81, 2005.
(Thomas, 2001)	G. Thomas, L. Johnson, S. Dow, and C. Stanford. “The design and testing of a force feedback dental simulator”. Computer Methods and Programs in Biomedicine, 64:53–64, 2001.
(Thurairajah, 2000)	L. Thurairajah, R. Pidaparti, and R. Meiss. “ Smooth muscle shortening: material model analysis and FE simulation”. Proceedings of the 22nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Volume 4, pp. 2564 - 2568, 2000.
(Tuchschmid, 2007)	S. Tuchschmid, M. Bajka, D. Szczerba, B. Lloyd, G. Székely, and M. Harders. “Modeling Intravasation of Liquid Distension Media in Surgical Simulators”. In Proc. of the conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), LNCS, vol. 4792, pp. 717-724, 2007.
(Uno, 1997)	S. Uno, and M. Slater. “The sensitivity of presence to collision response. In Proc. of IEEE Virtual Reality Annual International Symposium (VRAIS) (Albuquerque, New Mexico, pp. 95, 1997.
(Valtorta, 2005)	D. Valtorta, M. Hollenstein, A. Nava, V. Luboz, M. Lu, A. Choi, E. Mazza, Y. Zheng, and S. Cotin. “Mechanical characterization of soft tissue: comparison of different experimental techniques on synthetic materials”. Proceedings of the Fourth International Conference on the Ultrasonic Measurement and Imaging of Tissue Elasticity, p 94, 2005.
(van Walsum, 1997)	T. van Walsum, K. Zuiderveld, J. Chin-A-Woeng, B. Eikelboom, and M. Viergever. “CT-based simulation of fluoroscopy and DSA for endovascular surgery training”. In Proceedings CVRMed-MRCAS ‘97, Lecture Notes in Computer Science, Vol. 1205, pp. 273-282, 1997.
(Vigneron, 2004)	L. Vigneron, J. Verly, and S. Warfield. “On extended finite element method XFEM for modeling of organ deformations associated with surgical cuts”. In S. Cotin and D. Metaxas, editors, Medical Simulation: International Symposium (ISMS 2004), volume 3078 of Lecture Notes in Computer Science, pp 134–143, 2004.

(Wang, 2005)	F. Wang, E. Burdet, R. Vuillemin, and H. Bleuler. “Knot-tying with Visual and Force Feedback for VR Laparoscopic Training”. Engineering in Medicine and Biology Society (IEEE-EMBS), pp. 5778-5781, 2005.
(Wang, 2007)	F. Wang, L. Duratti, E. Samur, U. Spaelter, and H. Bleuler. “A Computer-Based Real-Time Simulation of Interventional Radiology”. In: Proc. of 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2007.
(Waters, 1992)	K. Waters. “A physical model of facial tissue and muscle articulation derived from computer tomography data. In Visualization in Biomedical Computing”, Proceedings of VBC, volume 574, 1992.
(Wells, 1996)	W. Wells, W. Grimson, R. Kikinis, and F. Jolesz. “Adaptive segmentation of MRI data”. IEEE Transactions in Medical Imaging, vol. 15, pp. 429–442, 1996.
(Wittek, 2005)	A. Wittek, R. Kikinis, S. Warfield, and K. Miller. “Brain shift computation using a fully nonliear biomechanical model”. In Proc. Medical Image Computing and Computer Assisted Interventions (MICCAI), pp. 583-590, 2005.
(Wu, 2005a)	X. Wu, K. Krissian, V. Luboz, S. Cotin, and S. Dawson. “Smooth Vasculature Reconstruction from Patient Scan”. Proceedings of the Virtual Environment Interactions and Physical Simulation workshop (VRIPHYS), November 2005.
(Wu, 2005b)	X. Wu, V. Pegoraro, V. Luboz, P. Neumann, R. Bardsley, S. Dawson, and S. Cotin. “New Approaches to Computer-based Interventional Neuroradiology Training”. Proceedings of 13th Annual Meeting, Medicine Meets Virtual Reality, 2005.
(Wu, 2007)	X. Wu, J. Allard, and S. Cotin. “Real-time Modeling of Vascular Flow for Angiography Simulation”. Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), Volume 4792, pp. 850-857, 2007.
(Yau, 2006)	H. T. Yau, L. S. Tsou and M. J. Tsai. “Octree-based Virtual Dental Training System with a Haptic Device”. In Computer Aided Design and Applications, Vol. 3, pp. 415-425, 2006.
(Yazici, 2005)	B. Yazici, B. Erdoğmuş, and A. Tugay. “Cerebral blood flow measurements of the extracranial carotid and vertebral arteries with Doppler ultrasonography in healthy adults”. Diagn Interv Radiol.; 1(4):195-198, 2005.
(Zhuang, 2000)	Y. Zhuang and J. Canny. “Haptic interaction with global deformations”. Proceedings of the IEEE ICRA conference, pp 2428-2433, 2000.
(Zhong, 2005)	H. Zhong, M.P. Wachowiak, and T.M. Peters. “Enhanced pre-computed finite element models for surgical simulation”. Proceedings of Medicine Meets Virtual Reality 13, pp 622–628, 2005.
(Zienkiewicz, 2000)	O. Zienkiewicz. “The Finite Element Method”. Volume Fifth Edition. Butterworth-Heinermann , 2000

(Zilles, 1995) C. Zilles and J. Salisbury. "A Constraint-Based God-Object Method For Haptic Display." In Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems, 1995.