

Creating an Infrastructure to Address HCMDSS Challenges

Peter Kazanzides and Russell H. Taylor
Center for Computer-Integrated Surgical Systems and Technology (CISST ERC)
Johns Hopkins University, Baltimore MD

Introduction

As leaders of the NSF Engineering Research Center for Computer-Integrated Surgical Systems and Technology (CISST ERC), we are keenly aware of the challenges in developing high-confidence software for complex medical devices. The mission of the CISST ERC is to develop computer-integrated surgical (CIS) systems that will significantly change the way surgical procedures are carried out. Specifically, we focus on families of systems that combine innovative algorithms, robotic devices, imaging systems, sensors, and human-machine interfaces to work cooperatively with surgeons in the planning and execution of surgical procedures. Our goal is to produce systems that will greatly reduce costs, improve clinical outcomes, and increase the efficiency of health care delivery. The following sections outline our positions on four of the topics identified in the November 2004 planning meeting. We believe that the topics are inter-related and therefore conclude with an integrated roadmap.

Enabling Technologies for Future Medical Devices

Much of our research involves the development of new technologies for medical devices and we believe that we are well qualified to present information in this area. Our most prominent research has been the development of a variety of surgical robots, including robots for orthopedics, ENT, urology and neurosurgery, but we also investigate many other enabling technologies, such as image segmentation, patient modeling, anatomical atlases, non-rigid registration methods, haptic interfaces, image overlay systems and navigation/tracking systems.

With respect to HCMDSS, we believe that the three most important challenges in this area are:

1. *Making systems that are efficient and easy to use in the operating room.* A common complaint about existing CIS systems are that they are too cumbersome to use, require additional setup time and often add to the total surgery time. They often do not provide sufficient sensory feedback and therefore reduce the surgeon's "feel".
2. *Establishing and maintaining the geometric relationships between different data sets, devices and the target anatomy.* CIS systems are often designed to execute a plan that is created using preoperative patient models, such as CT or MRI images. This requires a registration of the preoperative coordinate systems with the intraoperative (patient) coordinate systems (multiple preoperative and intraoperative coordinate systems may exist if multiple devices are employed). It further requires that the relationship be maintained during the surgical intervention by preventing motion/deformation of the target anatomy or by compensating for it. This is especially difficult when working with soft tissue and organs.
3. *Creating systems that are small enough for the desired surgical procedure.* The trend towards minimally invasive (or less invasive) surgery requires CIS systems that can work in small spaces. Perhaps the ultimate system was depicted in the 1966 movie *Fantastic Voyage*, in which the surgical team and a manipulation system are shrunk and injected into

the patient. The “shrunk” surgical team represents the need for intelligence at the operative site, which can more realistically be provided by telesurgery or possibly by artificial intelligence.

The above challenges require research in many areas, including mechanical and electrical design, but we believe that the three most important information technology research needs are:

1. *Tools to collect and analyze the steady stream of intraoperative data.* The introduction of computer-based systems into the operating room has led to an explosion in the amount of data that can be collected and displayed to the surgeon or made available for post-surgical analysis. In both situations, it is important to avoid information overload. There is a need for tools to collect the most pertinent information from multiple devices and present it in a comprehensible surgical display, as well as for tools that compile a more detailed amount of data into a comprehensive surgical record that can later be mined for information (e.g., to refine system designs to improve their usability).
2. *User interfaces that are context-aware and are suitable for the surgical environment.* Most OR procedures require sterile conditions and are therefore not well-suited for traditional user interfaces, such as using a mouse to select menu items. We found that surgeons often prefer a “hands-on” approach when interacting with a device; as a result, we created several robot systems using a “cooperative control” paradigm in which the surgeon and robot share control of the surgical tool. In this mode, the robot senses the tool forces applied by the surgeon and moves accordingly. The definition of “accordingly” may vary depending on the surgical context. In some cases, the robot should enforce safe (“no-fly”) zones and in other cases it should allow unconstrained motion. In our research, we have used “virtual fixtures” to enforce safe zones and we have used hidden Markov models to automatically detect the surgical context.
3. *Improvements in distributed processing and real-time communication.* Size constraints will force many systems to employ distributed processing. For example, it is impossible to place a small device into the body through a 4mm access port if the device requires a 50-conductor cable to transmit low-level I/O signals to an external computer. Such systems will require embedded processors connected via wired or wireless networks. The research need is for high-performance real-time communication between these distributed elements, which is especially important when compensating for changing conditions (e.g., target motion or deformation) and when providing high-fidelity sensory feedback to the surgeon (e.g., haptic feedback for telesurgery).

Foundations for Integration of Medical Device Systems/Models

With respect to HCMDSS, we believe that the three most important challenges in this area are:

1. *Transforming the current environment of standalone devices into an environment where devices can be interconnected.* Many medical devices are complex pieces of capital equipment and therefore it is difficult, expensive and risky for a product developer or researcher to duplicate functionality in order to create the complete system.
2. *Finding software toolkits to enable rapid prototyping and analysis of new ideas.* It is widely accepted that reuse of well-designed software components can reduce development time and increase quality. Few toolkits exist for medical device components and even fewer are designed with the appropriate focus on documentation and testing.

3. *Safety standards that can be applied to integrated systems.* Once it becomes possible to easily integrate different systems, there must be standards to ensure the safety of the integrated system.

What are the three most important information technology research needs?

1. *Broadly accepted interface standards for major components, such as robots, navigation systems and intraoperative imaging systems.* Presently, DICOM is the accepted standard for the exchange of medical images and the HL7 standard has been created for patient records. There is, however, no interface standard for many types of medical devices. A notable effort in this field is the IEEE 1073 standard, which does not yet address many new devices and is not yet fully accepted by manufacturers.
2. *Open source toolkits for major technology components.* Some open source toolkits already exist for medical image visualization and processing. In our own research, we identified the need for toolkits for medical robots and navigation/tracking systems. These systems also tend to require toolkits for registration between different coordinate systems.
3. *Tools that can predict the performance of an integrated system of sub-systems during the design phase.* These tools would allow developers to determine whether the desired system will meet the requirements before it is built, so that design changes can be made when their cost and schedule impact is still small. As an example, we have developed a theoretical framework and software implementation of a tool that predicts the total system accuracy (error) that would result from the integration of multiple subsystems.

Distributed Control & Sensing of Networked Medical Device Systems

With respect to HCMDSS, we believe that the three most important challenges in this area are:

1. *Safely handling errors and exceptions.* Error handling is especially difficult for devices that interact with the physical world and the challenge is even greater when the system contains distributed processing elements. If an erroneous physical action has been performed, “undo” is rarely an option. We need better designs (perhaps “design patterns”) for handling errors and exceptions in distributed real-time systems.
2. *Validation of correct system operation.* We also need better methods for validation of system performance, especially when testing the response of a system to fault conditions. This is a natural extension to the first challenge, which focuses on the design problem.
3. *Network hardware and software that enables real-time performance.* Many networks are designed to maximize throughput, but for a real-time system the key performance metric is often the message latency.

What are the three most important information technology research needs?

1. *Mechanisms for error propagation throughout the various software elements in the network.* It is especially challenging to deal with errors that occur asynchronously. An example is the failure of a motor amplifier for one robot joint. This failure should be propagated to the supervisory software so that it can take an appropriate safety action, such as stopping all other robot joint motions. A specific research need is a portable (with respect to programming language and operating system) method for handling “exceptions”.

2. *Systematic methods for handling error conditions.* In the industrial robotics field, it has been reported that 80-90% of the software statements are present to handle error or exception conditions. We believe that this is also true for many complex medical devices, especially since their environment is generally less structured than a typical manufacturing environment. If error handling code can be reduced via systematic methods, it would lead to a drastic reduction in the size and complexity of medical device software, thereby reducing the cost of development, qualification and maintenance. We have done some work in this area, primarily during the development of the ROBODOC® System.
3. *Middleware that can provide low latency communication.* There are middleware efforts such as RT-Corba and DDS to address distributed real-time systems. This field requires further maturation.

Embedded, Real-Time, Networked System Infrastructures for MDSS

We interpret this category to include all types of networked system infrastructures, not just those that are “embedded, real-time.” With respect to HCMDSS, we believe that the three most important challenges in this area are:

1. *Defining the requirements.* We do not believe that the requirements are fully known at this time. One of our research goals has been to evaluate different middleware options for the integration of devices such as surgical robots, intraoperative imaging systems and navigation systems. We created a requirements document to drive our evaluation, but it is far from complete and we suspect that a complete requirements document does not exist.
2. *Achieving true “plug and play” capabilities, where the integrated system can automatically determine whether the newly attached device is suitable.* There are some regulatory challenges here as well, since current regulations require any integrated system to be cleared by the FDA. In the future, it may be feasible to allow some amount of “on the fly” integration, provided that there are strict safeguards on interoperability of devices.
3. *Defining an open architecture and middleware that can be adopted by the research community and by industry.* This does not yet exist. Some developers use a client/server model with industry standard middleware such as CORBA or SOAP. More commonly, developers use closed architectures with proprietary interconnection mechanisms.

What are the three most important information technology research needs?

1. *Documenting “use cases” for medical device networks.* We believe that “use cases” (example scenarios) can help with the definition of the requirements.
2. *Developing a taxonomy for medical devices.* The universe of medical devices must be categorized and then for each category, it is necessary to define the set of all possible features. This is not an easy task – it is especially difficult to figure out where to draw the line. For example, is the set of all medical robots a single category or are there several categories depending on number of degrees of freedom, kinematics, etc.?
3. *Identifying a solution that can satisfy both real-time and high-level interface requirements.* In our experience, high-level integration of major subsystems is most conveniently implemented with middleware that supports a client/server architecture. On the other hand, real-time data distribution is often better implemented with publish/subscribe middleware, especially if multicast capabilities are important.

Integrated Roadmap

We believe the primary goal should be to increase the integration of new devices, sensors and algorithms into working systems. Progress in the field will be hampered if developers must expend years of effort to build the testbeds that they require to integrate and test new ideas. For example, a researcher could spend years developing a master/slave robot system just to be able to test new haptic feedback algorithms. Our integrated roadmap, therefore, stresses the development of standardized interfaces, open architectures and component toolkits to facilitate research and development in computer-integrated surgery. Essentially, we propose a roadmap that focuses on the development of an infrastructure that enables researchers and developers to more quickly and effectively address the above challenges.

Over the next 5 years:

- Foster the development of documented and tested open source toolkits for research with new medical devices such as robots, haptic interfaces and navigation systems. The toolkit set should also include “foundation” toolkits that solve generic problems such as logging of intraoperative data.
- Encourage device manufacturers to provide open research interfaces to their devices so that researchers can implement new ideas on existing hardware.
- Document “design patterns” or “best practices” for handling errors and exceptions in distributed real-time medical device software.
- Document medical device “use cases” that can lead to the definition of requirements for networked system infrastructures.
- Create tools to simulate (or prototype) new medical devices and predict their performance. These tools can be used to uncover problems in the design phase, where the cost of correction is the lowest.

Over the next 10 years:

- Develop tools to analyze intraoperative data. This could include virtual reality simulators that enable researchers to “debug” a surgical procedure (consider, for example, the ability to utilize software debugging concepts such as “single step”).
- Evolve the open research interfaces into standards for medical device interfaces. These standards must be supported by industry, researchers and regulatory agencies.
- Define standards for safety design in distributed medical devices. These standards should capture the “design patterns” and “best practices” for error handling, but should have a broader scope.
- Define the middleware requirements and evaluate existing middleware against these requirements (and “use cases”). It may be possible to eventually standardize on a single middleware package.

Biographies

Peter Kazanzides received the Sc.B., Sc.M., and Ph.D. degrees in electrical engineering from Brown University in 1983, 1985, and 1988, respectively. His dissertation focused on force control and multiprocessor systems for robotics. He began work on surgical robotics in March 1989 as a postdoctoral researcher at the IBM T.J. Watson Research Center with Dr. Russell Taylor. Dr. Kazanzides co-founded Integrated Surgical Systems (ISS) in November 1990 to commercialize the robotic hip replacement research performed at IBM and the University of California, Davis. As Director of Robotics and Software, he was responsible for the design, implementation, validation and support of the ROBODOC® hardware and software. Dr. Kazanzides joined the NSF Engineering Research Center for Computer-Integrated Surgical Systems and Technology at Johns Hopkins University in December 2002 as Chief Systems and Robotics Engineer. He currently holds an appointment as an Assistant Research Professor of Computer Science at Johns Hopkins University.

Russell H. Taylor received a B.E.S. degree from The Johns Hopkins University in 1970 and a Ph.D. in Computer Science from Stanford in 1976. He joined IBM Research in 1976, where he developed the AML robot language. Following a two-year assignment in Boca Raton, he managed robotics and automation technology research activities at IBM Research from 1982 until returning to full time technical work in late 1988. From March 1990 to September 1995, he was manager of Computer Assisted Surgery. In September 1995, Dr. Taylor moved to Johns Hopkins University as a Professor of Computer Science, with joint appointments in Radiology and Mechanical Engineering. He is also Director of the NSF Engineering Research Center for Computer-Integrated Surgical Systems and Technology. Dr. Taylor has a long history of research in computer-integrated surgery and related fields. In 1988-9, he led the team that developed the first prototype for the ROBODOC® system for robotic hip replacement surgery and is currently on the Scientific Advisory Board of Integrated Surgical Systems. At IBM he subsequently developed novel systems for computer-assisted craniofacial surgery and robotically-augmented endoscopic surgery. At Johns Hopkins, he has worked on all aspects of CIS systems, including modeling, registration, and robotics in areas including percutaneous local therapy, microsurgery, and computer-assisted bone cancer surgery. He is Editor Emeritus of the IEEE Transactions on Robotics and Automation, a Fellow of the IEEE and the AIMBE, and a member of various honorary societies, panels, editorial boards, and program committees.

Contact Information

Peter Kazanzides
e-mail: pkaz@cs.jhu.edu
phone: 410-516-5590

Russell H. Taylor
e-mail: rht@cs.jhu.edu
phone: 410-516-6299