

C O N T I N U I N G

E D U C A T I O N

2.5 ANCC
Contact Hours

Development of a Simulated Smart Pump Interface

BETH L. ELIAS, PhD, MS
JACQUELINE A. MOSS, PhD, RN, FAAN
ALAN SHIH, PhD
MARCUS DILLAVOU

Medical devices, such as smart intravenous medication infusion pumps, are incorporating computerized information technology into their designs to support clinical decision making, reduce the incidence of preventable error, and improve patient outcomes.¹ Medical device user interfaces, therefore, are increasing in complexity and are in need of design and interface usability testing with clinicians in a clinically accurate setting.² Observing the use of smart pumps and other smart medical devices in the clinical setting is not the optimal method to investigate the effect they may have on clinical practice. Introducing and manipulating device technology in the clinical setting can have unknown and adverse consequences and may require many rigorous observations to uncover design flaws or operations that can potentially induce or correct operator error.³ This in situ method increases the chance of introduced error and harm to patients and limits designers' ability to fully develop and measure the effect of redesigns needed to decrease these errors.⁴

Simulating smart pumps, other medical devices, and information technology interfaces allows the researcher to introduce a variety of medical scenarios and potentially error-producing design features (ie, interface layout, menu structure, and drug library complexity) without the risk of harming actual patients.⁵ This article describes the first phase in a study designed to provide an understanding of how medical device design influences how nurses interact with and use technology-enabled devices. It was aimed at developing a simulation technology that can provide easy access to larger user groups for testing of new interface designs, training in the use of smart device interfaces, and for evaluation purposes.

Medical device user interfaces are increasingly complex, resulting in a need for evaluation in clinically accurate settings. Simulation of these interfaces can allow for evaluation, training, and use for research without the risk of harming patients and with a significant cost reduction over using the actual medical devices. This pilot project was phase 1 of a study to define and evaluate a methodology for development of simulated medical device interface technology to be used for education, device development, and research. Digital video and audio recordings of interface interactions were analyzed to develop a model of a smart intravenous medication infusion pump user interface. This model was used to program a high-fidelity simulated smart intravenous medication infusion pump user interface on an inexpensive netbook platform.

KEY WORDS

Simulation • Smart devices • Smart infusion pump • User satisfaction

BACKGROUND

Intravenous medication administration has been cited as one of the most preventable sources of error risk to patients.^{6,7} In studies examining the sources of these errors,

Author Affiliations: Department of Community Health, Outcomes and Systems (Dr Elias) and School of Nursing, Department of Adult Health (Dr Moss), University of Alabama at Birmingham; and Enabling Technologies Laboratory, Mechanical Engineering, School of Engineering, University of Alabama at Birmingham (Dr Shih and Mr Dillavou).

The authors gratefully acknowledge receiving intramural funding in the form of a dean's Scholar Award from the University of Alabama at Birmingham School of Nursing for this study.

The authors have disclosed that they have no significant relationship with, or financial interest in, any commercial companies pertaining to this article.

Corresponding author: Beth L. Elias, PhD, MS, School of Nursing, Department of Community Health, Outcomes and Systems, University of Alabama at Birmingham, 1702 2nd Ave S, Birmingham, AL 35294 (blelias@uab.edu).

DOI: 10.1097/CIN.000000000000016

it was found that they included the complexity of the medication administration process, dose calculation, and inexperienced or distracted healthcare providers.^{8,9} Medical devices such as smart intravenous medication administration pumps can provide clinical decision support to nurses at the bedside to reduce these errors but are being implemented with mixed results perhaps due to lack of understanding of how they function in the clinical setting.^{1,10}

With standard infusion pumps, the fluid volume and infusion rate are entered into the pump interface by the nurse: milliliters per hour, for example. The pump then controls the infusion to meet these criteria. These standard pumps are themselves an improvement over the older thumbwheel control of drip rate, which required nurses to use simple pressure on the tubing by moving a wheel to open or restrict medication flow.¹¹ Neither device is capable of providing decision support to help prevent known causes of error.

Smart medical devices, specifically smart intravenous medication infusion pumps, incorporate computer hardware and software that is designed to reduce known sources of preventable error.¹¹ Smart pumps carry an on-board drug library that consists of a list of intravenous medications, the concentrations of medication available, therapeutic dose ranges, and dangerous dose ranges. Nurses program the smart pump interface, selecting the drug, concentration, and volume to be infused, and entering patient data such as weight. Smart pumps use the data to calculate the rate and duration of the infusion.

The pump software also checks the entered data against the therapeutic dose range, alerting the nurse if a non-therapeutic but nondangerous dose is programmed (soft limit). The nurse is prompted to verify that the entered data are correct before proceeding. An alert will also be triggered if a dangerous dose has been entered (hard limit), forcing the nurse to reprogram the infusion. Smart pumps are also wireless Internet enabled, which allows drug libraries to be updated via remote access. Medication administration logs and error logs can similarly be downloaded on a regular basis to a central system for analysis.

To realize the potential of these smart medical devices, we must clearly understand how they are implemented and used in the clinical setting. For example, one area of concern is the order and size of the on-board drug libraries, primarily due to differences in drugs and dosing among hospital units.¹² Also of concern is the design of the pump's keypad and display. The poor readability of small, inadequately lit displays that use small text fonts can result in increased time required to initiate an infusion. These design flaws can result in incorrect entry of dosing information and introduce new causes of error into the medication administration process.¹³

High-fidelity simulation is an effective means of providing clinically accurate educational experiences for nursing students¹⁴ and understanding the functioning of strong

healthcare teams.¹⁵ Simulation can contribute to the reduction of preventable errors by allowing practice to take place in a clinically accurate, low-risk setting. However, including real medical devices in training is often costly, and it can be impractical to issue devices to students and nurses.

Simulation is increasingly used as a tool in healthcare to evaluate new technologies and processes as they are introduced.¹⁶ Computerized modeling and forecasting techniques using real or simulated data have been successfully used in the past to study issues related to medication administration error.^{17,18} These modeling simulations, however, cannot provide information about how real people might interact with or develop workarounds for medical devices such as smart pumps.

OBJECTIVES

This article describes the first phase of a two-phase study aimed at defining a methodology for development of simulated medical device interface technology to be used for education, training, development, validation, and research purposes. The objectives for this phase consisted of (1) development of a map of the features and functions used for programming medication administration using a smart pump, (2) development of a model of the user interface layout of a smart pump, and (3) programming a simulation of smart pump features, functions, and user interface on an inexpensive netbook computer. The methodology was used to develop a simulated smart pump interface technology (SSPT) that reproduces the visual, audio, and tactile experience of smart intravenous medication administration pump interface use.

METHODS

Three levels of smart pump interface interaction were recorded using digital video and audio capture. The smart intravenous medication administration pump, a Hospira Symbiq Infusion System (Hospira, Lake Forest, IL), was on loan from an acute care hospital and was normally used in training for unit nursing staff. The digital video and audio sessions were analyzed through an iterative process and documented in Microsoft Excel (Microsoft, Redmond, WA). The level of granularity of the documentation was refined through the three levels of digital video and audio recording until an event-driven user interface model emerged. This model was then used to describe a finite state machine (FSM) in which the interface pattern of events, states, and state transitions was fully representative of the actual device. The state machine can be defined as a set of states, transitions

between those states, and some final states that capture all possible features and functions of the device.

An example of a state for a smart pump is the presentation of the screen that allows selection of the A or B channel for a two-channel pump designed to infuse two medications at the same time. One transition into this state would be turning the pump on, by the action of pushing the on button. A transition out of this state would be selecting A or B channel, by the action of pressing A or B on the touchscreen. The concept of correct and error transitions comes from what is appropriate for the workflow. For example, when programming a medication infusion for channel A, pressing B would then be an error, although allowed by the pump software.

Exploring Pump Interaction Recording Session

The first pump interaction level involved a complete exploration of the features and functions of the device (EPI) by the principal investigator (PI), moving through all options and views. A digital audio recording was included as part of the videotape of activities to capture a spoken description of actions and reactions during the exploration. The audio capture provides a narration of the activity, giving detailed impressions of the interaction with the device interface by the user.

In a Microsoft Excel worksheet, the states and transition triggering events were documented. Each included the previous state, current state, current state entry operation (transition trigger to enter current state), a description of the current state, potential correct operations to transition to another state, and potential error operations that would exit the current state. The digital video and audio were reviewed in an iterative fashion in increasingly fine granularity until a detailed model of the interface emerged. Once no further refinement could be made from the first recording session, the second recording session was analyzed to add the nursing workflow for medication administration. For example, when programming medication infusion, the first step is that the appropriate pump channel must be selected, then the correct medication must be selected from the drug library, which is followed by selecting the correct drug concentration. Patient information such as weight also needs to be entered for some drugs after the drug and concentration have been entered. The nursing workflow identifies the path through the states and the end point.

Novice Recording Session

The second pump interaction level recorded was attempted medication administration programming for heparin and insulin by a novice (NOV), the PI, to trigger alerts for

dangerous dosing and dosing outside the normal therapeutic range as well as states triggered by programming errors and misplaced touch. This NOV recording was made after the PI attended the required in-service for the smart pump at the lending hospital. The user manual was also reviewed, as it would have been by a novice user after the in-service.

A second Excel worksheet was created by cutting and pasting states from the EPI worksheet as needed and adding documentation for new states that were triggered by the NOV programming. Again, a think-aloud protocol added information in the form of narration and reaction to the new error and alert states as well as reaction to the sounds made by the pump while in those states. Transitions were also described, as was the tactile experience of interacting with the touchscreen, both correctly and incorrectly. As with the EPI session, the digital recording was analyzed in an iterative fashion until no further detail emerged. Any corrections to the detail used from the EPI session were reflected in the EPI worksheet.

Medication Administration Programming Recording Session

The third pump interaction level digital video and audio session recorded expert smart pump interface programming for heparin and insulin by an RN who had hospital training and clinical experience using the smart pumps. This session captured the clinically accurate nursing process workflow for intravenous medication administration programming (MAP) by an experienced nurse (EXP). The speed and timing of this programming were captured in this third recording, as was the tactile understanding of how to successfully interact with the interface that came from experience. The exactness of fingertip placement, the pressure that must be exerted, and the length that pressure must be maintained on the touchscreen are all part of the tactile understanding that nurses develop with experience using the pump touchscreen interface.

A third Excel worksheet was created, again by cutting and pasting from the EPI and NOV worksheets as needed to describe the expert programming. A think-aloud protocol narrated the interaction in the words of the RN. When necessary, detail in the EPI and NOV documentation was clarified and reflected in the EPI and NOV worksheets. No additional states or transitions needed to be described; however, the rapid speed of touchscreen interaction at times did trigger an audible error beep when the touch was too fast and at times not firm enough to be correctly interpreted by the pump.

With the completion of the analysis of the third recording session, we were able to document and completely describe the MAP from the states and transitions that had emerged from the EPI and NOV analysis. It was at

this point that the fine-grained FSM was judged to be complete. Several states, with some detail, from the FSM are shown in Table 1.

The FSM was then used to guide the development of a Digital Video and Audio Finite State Machine (DVFSM), which allowed us to represent the FSM in a visual and aural form, showing each state and transition in real time as experienced by the nurse or PI

Digital Video Editing

Digital recording frame numbers were added to each state and transition for the three worksheets (EPI, NOV, MAP) in the FSM. Using Adobe Premier Pro CS4 (Adobe Systems, San Jose, CA), the digital video and audio recordings were edited as three video clips, one for each pump programming session. Video chapter marks were added for each state and transition in the FSM and were named the same as in the FSM for consistency between the FSM and the DVFSM. Extraneous video and audio were removed for clarity; audio volume was enhanced and static removed. Particularly important alert and error sounds were also given chapter marks, as were particularly clear examples of tactile interaction with the touchscreen. These additional chapter marks were also described in the FSM as within-state features.

A high level of consistency was maintained between the states and transitions captured in the FSM and DVFSM in two ways, using the same naming convention and including the video frame numbers of the chapter marks in the FSM. This consistency allowed for a high level of interaction and moving between the two, an aspect of the methodology that we felt was necessary for rapid

prototyping of the software framework. Reading a detailed description of a state could be quickly followed by viewing a video clip which resulted in the development of a higher-fidelity beta version of the device interface than would normally be achieved.

The DVFSM was then authored (that is, saved with navigation, menus, and visual elements) onto a DVD. The DVFSM had a primary menu listing the three interaction sessions, EPI feature review, NOV insulin and heparin, and MAP. Submenus that included chapter marks for each state, transition, and within-state features, as shown in Figure 1, then further delineated each interaction level. This organization mirrored the organization of the FSM and allowed the rapid prototyping of the simulated interface.


While the FSM captures the features and functions of the device as a set of states and transitions, the DVFSM adds the visual and aural experience of interacting with the smart pump interface as well as demonstrating the tactile sensitivity of the pump touchscreen.

Software Framework

This FSM and DVFSM provide the specifications for programming a simulated SP interface. The interface was developed using common Web technologies including Javascript, HTML, and cascading style sheets (CSS). The system used a custom Web browser based on WebKit, an open source browser engine (<http://www.webkit.org/>). This was because the netbook was required to function as a medical device, and full control of the system was necessary. However, the simulator program can also be run over the Web in any Web browser for remote training.

Table 1

Smart Pump Finite State Machine



Prestate	State	Entry Operation for Transition	Description	Alternate States (Potential Errors)
Pump on	Pump off	Not applicable	Image of pump housing	NA
Pump off	Pump on	Press on button	Initial pump screen	Press off button: no effect
Pump on	Select channel A	Two-channel pump (A and B), selection of A channel: press on letter A	Channel A screen	Select channel B
Complete channel A screen	Infusion: select channel A	Select channel A	Post-patient info to begin drug selection and infusion programming	NA Default display is for basic tab
Infusion: select channel A	Select infusion screen	Press infusion: select infusion arrow	Display drug library pop-up	The other buttons if pressed would just do the three-beep error
Select drug library	Select heparin	Press: on scrollbar to scroll down drug library to heparin select: heparin	Selecting heparin from drug list	Select channel B, select drug other than heparin, press cancel
Complete channel A screen	Infusion: select channel A	Select channel A	Post-patient info to begin drug selection and infusion programming	NA default display is for basic tab

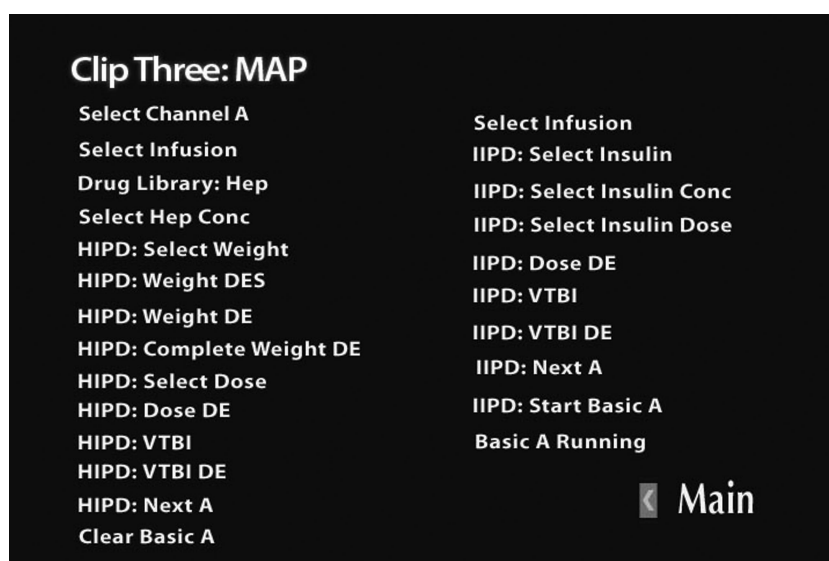


FIGURE 1. Expert programming submenu showing organizational structure of DVD.

Using the FSM, a state machine was developed in JavaScript, an open source programming language. This code managed all the states of the simulator and linked the data model and data calculation algorithms. The interface was developed using CSS, a common Web technology, and was based on the actual interface of the Hospira Symbiq, using similar colors, icons, and button placement for effective training and analysis. By using proven Web technologies, we were able to rapidly develop the simulator while keeping it generic enough to support other modes.

While much of the workflow of the simulator was based on the FSM, the simulator also needed to hold a Drug Library Database. The Drug Library, taken from an actual medial/surgical hospital unit, was imported from a comma-separated values file, stored on disk, and loaded into an in-memory database. The Drug Library stored all required information about the drugs for performing calculations and presenting information to the user, including names, units, and upper and lower limits, along with dosing amounts.

The simulator software was developed using a rapid prototyping method. Using this method allowed the developers and research team to make changes to the simulator quickly and often until the simulator fully complied with specifications.

Software and Hardware Platform

The simulated smart pump interface was developed using an ASUS eee PC Touch T91MT Netbook (ASUS, Taipei, Taiwan), built on a customized Linux operating system. This category of computer systems offers an ideal platform to develop various medical device simulators in a very cost-effective manner. Tablet computers provide sufficient computing power, excellent graphics quality, and touchscreens that are close in size and tactile experience to the actual medical devices, allowing for an authentic simulation of the device interface. The ASUS eee also allows the touchscreen to be rotated to a position atop the keyboard, removing it from the view of the user. The netbook is shown in Figure 2.

THE SIMULATED SMART MEDICATION INFUSION PUMP INTERFACE

The initial version of the SSPT software, and hardware, was verified against the DVFSM, by the PI, by simultaneously moving through SSPT states while viewing the



FIGURE 2. ASUS eee PC Touch T91MT Netbook.



FIGURE 3. Actual smart pump interface on the left, SSPT on the right.

same states and transitions in the digital video and audio. Discrepancies were noted and addressed by revision of the SSPT programming. Once a beta version was available, the SSPT was reviewed and tested by three expert nurses. Comments were recorded and reviewed by the PI and development team. Each comment was designated as either “no action needed” or “action needed” with specific follow-up steps documented. After several rounds of expert review, the beta SSPT was then tested, by programming heparin and insulin medication administration, in parallel with an actual smart pump by a research technician, who was a fourth-year undergraduate nursing student. The research technician had also received training in the use of the pump in the clinical practicum setting. No differences were noted between the SSPT and actual smart pump. The SSPT was then felt to be a high-fidelity simulation of the actual smart pump interface. The actual pump screen and the simulated screen are shown in Figure 3.

LIMITATIONS

One limitation of using a netbook platform is the need to capture keystrokes that are interpreted as commands by the operating system software, which may interfere with the netbook’s simulation of the smart pump user interface by initiating actions that are outside the functions of the simulator. For example, dragging fingertips on a touchscreen can be interpreted by the operating system as a command to enlarge or copy a section of the screen. Common finger touch patterns that trigger any such commands will be identified by the next-steps study, so that they can be trapped and suspended by the simulation software.

An additional limitation of this pilot study is the small number of subjects used to map the features and functions of the actual smart pump. As with many pilot studies with small amounts of funding, our goal was to develop a

method that could then be used in a larger study to more fully develop the simulation. In the next phase of the study, we intend to use a larger number of subjects with varied skill sets to ensure that we can simulate all possible errors and also to explore workarounds.

DISCUSSION

The goal of describing a methodology to simulate medical device interfaces that included tactile, audio, visual, and workflow representations was largely successful. The simulated smart medication infusion pump interface developed from this methodology provided a simulated user interface experience for programming heparin and insulin infusions that was used in the second phase of this study comparing the SSPT to an actual smart pump.¹⁹ The second phase was intended to evaluate the SSPT and to help identify aspects of the interface programming where the SSPT needs to be improved, as well as areas where the actual smart pump could benefit from redesign to reduce the number of errors generated during medication administration programming. In this comparison study with fourth-year nursing students, with no prior pump experience, the pilot simulated smart pump interface was compared with the actual pump interface for programming the infusion of heparin and insulin. The simulated pump did not significantly differ from the actual pump except for a higher number of errors generated by use of the number pad to enter weight and dose data. This indicates an area where the simulation can be improved to increase fidelity.

Realizing the full benefit of smart medical device technology depends on accurate and complete evaluation of the physical aspects of medical devices as well as their interfaces. Simulated medical device interfaces, which can provide an

accurate visual, tactile, and audio experience, will allow for this evaluation in a safe setting and will also allow for rapid, iterative, and low-cost usability testing as new technologies emerge or as existing technologies evolve. In addition, this methodology will provide a model that allows for the simulation of multiple medical device interfaces on a single low-cost netbook platform.

Medical devices are essentially event-driven systems that can assume a limited number of a priori determined values and that change from one state to another through defined transitions. This study can contribute to the body of knowledge by describing a methodology for the simulation of smart medical device technologies, providing a means for including a wide variety of simulated devices in device design, evaluation, simulation research studies, and healthcare education and training. Given the fact that applications on smartphones, netbooks, and pad-based computer devices are ubiquitous in today's society, it provides us with an ideal environment to develop and deploy such medical simulators.

REFERENCES

1. Murdoch LJ, Cameron VL. Smart infusion technology: a minimum safety standard for intensive care? *Br J Nurs*. 2008;17(10):630–636.
2. Cafazzo JA, Trbovich PL, Cassano-Piche A, et al. Human factors perspectives on a systemic approach to ensuring a safer medication delivery process. *Healthc Qual*. 2009;12 Spec. no. patient:70–74.
3. Crimlisk JT, Johnstone DJ, Sanchez GM. Evidence-based practice, clinical simulations workshop, and intravenous medications: moving toward safer practice. *Medsurg Nurs*. 2009;18(3):153–160.
4. Strom BL, Schinnar R. Evaluating health information technology's clinical effects. *LDI Issue Brief*. 2011;16(4):1–4.
5. Saleem JJ, Patterson ES, Militello L, et al. Impact of clinical reminder redesign on learnability, efficiency, usability, and workload for ambulatory clinic nurses. *J Am Med Inform Assoc*. 2007;14(5):632–640.
6. Bates DW. Preventing medication errors: a summary. *Am J Health Syst Pharm*. 2007;64(14 suppl 9):S3–S9; quiz S24–S26.
7. Hicks RW, Becker SC. An overview of intravenous-related medication administration errors as reported to MEDMARX(R), a national medication error-reporting program. *J Infus Nurs*. 2006;29(1):20–27.
8. McDowell SE, Mt-Isa S, Ashby D, Ferner RE. Where errors occur in the preparation and administration of intravenous medicines: a systematic review and Bayesian analysis. *Qual Saf Health Care*. 2010;19(4):341–345.
9. Moss JA, Berner E, Bothe O, Rymarchuk I. Intravenous medication administration in intensive care: opportunities for technological solutions. AMIA Annual Symposium Proceedings/AMIA Symposium. *AMIA Symp*. 2008:495–499.
10. Trbovich PL, Pinkney S, Cafazzo JA, Easty AC. The impact of traditional and smart pump infusion technology on nurse medication administration performance in a simulated inpatient unit. *Qual Saf Health Care*. 2010;19(5):430–434.
11. Elias BL, Moss JA. Smart pump technology: what we have learned. *Comput Inform Nurs*. 2011;29(3):184–190.
12. Wetterneck TB, Skibinski KA, Roberts TL, et al. Using failure mode and effects analysis to plan implementation of smart i.v. pump technology. *Am J Health Syst Pharm*. 2006;63(16):1528–1538.
13. Kirkbride G, Vermace B. Smart pumps: implications for nurse leaders. *Nurs Adm Q*. 2011;35(2):110–118.
14. Alinier G, Hunt B, Gordon R, Harwood C. Effectiveness of intermediate-fidelity simulation training technology in undergraduate nursing education. *J Adv Nurs*. 2006;54(3):359–369.
15. Klipfel JM, Gettman MT, Johnson KM, et al. Using high-fidelity simulation to develop nurse-physician teams. *J Contin Educ Nurs*. 2011;42(8):347–357.
16. Alsos OA. Attention and usability issues in mobile health information systems at point-of-care. *Stud Health Technol Inform*. 2008;136:877–878.
17. Anderson JG. A focus on simulation in medical informatics. *J Am Med Inform Assoc*. 2002;9(5):554–556.
18. Wakefield DS, Wakefield BJ, Borders T, Uden-Holman T, Blegen M, Vaughn T. Understanding and comparing differences in reported medication administration error rates. *Am J Med Qual*. 1999;14(2):73–80.
19. Elias BL, Moss JA, Dillavou M, Shih A, Azuero A. Evaluation of nursing student perspectives of a simulated smart pump. *Clin Simulation Nurs*. 2013. In press.

For more than 31 additional continuing education articles related to computers in nursing, go to NursingCenter.com/CE.