

# IV Smart Pumps and Error-Prone Programming Tasks: Comparison of Four Devices

Karen K. Giuliano

With an estimated 90% of hospitalized patients receiving intravenous (IV) medications via infusion pumps, these devices are among the most ubiquitous technologies in healthcare.<sup>1</sup> During the previous two decades, IV smart pumps with built-in dose-error reduction systems (DERSs) have become widely accepted as a standard of care for addressing the problem of infusion-related medication errors. Before IV smart pumps were developed, all pump programming required users to manually calculate the rate of the infusion, then input the desired infusion rate into the pump. Because many different units of measurement are used in the administration of IV medications, dosage calculations often are complex, increasing the likelihood of user error.<sup>2</sup>

In contrast, IV smart pumps have built-in drug libraries and a DERS. Users select the desired medication from an approved list and input the required patient information, then the IV smart pump calculates the infusion rate. Various prompts and alerts also are integrated into the DERS, providing a warning to users if the programmed dose is outside of acceptable dosing limits. Data indicate that the use of IV smart pumps has been associated with reductions in medication error rates; however, the devices have not eliminated the potential for error.<sup>3-5</sup> Furthermore, current data do not indicate that the use of IV smart pumps has had a measurable impact on decreasing the most serious errors: adverse drug events (ADEs).<sup>3,6,7</sup> Considering the frequency and importance of IV medication administration in acute care, it is noteworthy that little work has been done regarding why ADEs have not been decreased or if differences exist among IV smart pump types.

To complete IV smart pump programming, users are required to interact with the pump at the user interface. The user inter-

face includes all parts of the pump with which the user must interact to complete the desired medication administration task, including interaction with both hardware and software. Much of the user interaction for IV smart pump programming is driven by the DERS software; therefore, the manner in which data and prompts are organized and presented is an important aspect of the user interface.

Currently, with no best practice standards, user interfaces vary markedly among the different IV smart pumps. Much of the empiric data on IV smart pump usability to date has been grounded in human factors principles, using a wide array of methods including heuristics, qualitative interviewing, usability simulation, cognitive assessment testing, clinical observation and real-time clinical evaluations.<sup>8-15</sup> However, available data comparing the workflow and usability for the user interfaces of different smart pumps are scarce.<sup>9</sup> Thus, little is known about what can and should be required by manufacturers to help make their products safer and easier to use.

## IV Smart Pump Medication Administration Tasks

IV medication administration is a complex, multistep process that provides numerous opportunities for error, especially when multiple infusions are being administered.<sup>16-18</sup> Although IV medication infusion is a high-risk activity, research has identified multiple infusions, weight-based infusions, secondary infusions, and IV boluses as being particularly high risk and error prone.<sup>18-20</sup> These tasks place additional cognitive demands on users, are not well standardized within hospital protocols and among IV smart pump user interfaces, and have many associated failure modes.<sup>18</sup> In addition, the failures for these tasks are not easily

*Karen K. Giuliano, PhD, RN, FAAN, MBA, is an associate professor of nursing at Northeastern University in Boston; an external faculty nurse scientist in the Yvonne L. Munn Center for Nursing Research at Massachusetts General Hospital in Boston; and a nurse scientist at the Center for Nursing Research and Advanced Practice at Orlando Health in Orlando, FL. Email: kkgiuliano96@gmail.com*

detected.<sup>18</sup> This study sought to take a closer look at two of these programming tasks: secondary infusions and weight-based infusions, in the context of large-volume IV pumps (LVPs).

### Secondary Medication Infusions

Secondary medication infusion by LVP is a common method for administering IV medications ordered for one-time or intermittent dosing, especially IV antibiotics. Secondary administration is designed to allow the primary continuous infusion to pause during the secondary infusion and resume automatically after the secondary infusion is complete. Because much of the setup and medication administration process for secondary medication administration must be managed manually, secondary infusions are prone to user errors.<sup>18</sup>

For example, most of the LVPs in current clinical use require a sufficient differential in the IV bag height between the primary and secondary lines, with the secondary infusion positioned higher than the primary infusion. This bag height differential is required for most IV smart pumps to allow the secondary medication to infuse completely before the primary infusion resumes.<sup>18</sup>

Two additional common errors with secondary infusion include 1) the use of tubing without a primary line back-check valve to prevent the secondary medication from infusing into the primary bag instead of into the patient and 2) neglecting to release the roller clamp on the secondary tubing, thereby preventing the secondary medication from infusing at all.

The following example of an error associated with secondary medication infusion was taken from the Food and Drug Administration (FDA) Manufacturer and User Facility Device Experience (MAUDE) database in 2016. A secondary infusion of piperacillin/tazobactam was ordered to infuse over four hours. However, when the nurse checked the infusion 1.5 hours later, the piperacillin/tazobactam bag was empty. After checking the setup and flow rate, the nurse noticed that the piperacillin/tazobactam had infused at 100 mL/h, which was the rate for the primary infusion. An internal review at the hospital attributed the overinfusion to user

error; however, this is a common problem with secondary infusion that may have resulted from an inadequate bag height differential between the primary and secondary infusion bags. Although some IV smart pumps in clinical use currently have technology that does not require a bag height differential for secondary infusions, this error occurred on an IV smart pump without that capability.

### Weight-Based Infusions

Medications that are administered as continuous infusions with dosage calculations dependent on patient weight are referred to as weight-based infusions. Many high-alert medications used in acute and critical care are delivered as weight-based infusions. Examples include dopamine, dobutamine, epinephrine, and sodium nitroprusside. Because weight-based infusions tend to be high-potency medications, IV medication errors that occur during weight-based infusion are more likely to cause a serious ADE.

The following example of a medication error during a weight-based infusion was taken from the MAUDE database in 2016. A high-concentration bag of norepinephrine bitartrate (16 mg/250 mL) was being infused on a patient in the intensive care unit (ICU). Before surgery, the norepinephrine bitartrate was discontinued and the bag and tubing were discarded. During surgery, the medication infusion was restarted but with a standard concentration (4 mg/250 mL) that was still infusing upon postoperative readmission to the ICU. When the norepinephrine bitartrate bag from the operating room was empty, the ICU nurse replaced it with a high-concentration bag of the medication (16 mg/250 mL) but did not adjust the programming parameters, which were set for the standard concentration, resulting in an infusion that was four times greater than appropriate. Approximately 12 hours later, the patient coded. The patient was successfully resuscitated, with vital signs returning to baseline status.

The purpose of the current study was to compare the required programming workflows of three currently available LVPs with a precommercial prototype infusion pump for

programming the high-risk tasks of secondary medication administration and weight-based infusions. The three LVPs included in the study were Alaris (Becton Dickinson, Franklin Lakes, NJ), Sigma Spectrum (Baxter, Deerfield, IL), and Plum A+ (ICU Medical, San Clemente, CA). These three manufacturers represent approximately 88% of the LVPs currently in clinical use in U.S. hospitals, with Alaris being the most widely used device.<sup>21</sup> The LVP used for comparison was the Ivenix Large-Volume Pump, which is a component of the Ivenix Infusion System (Ivenix, North Andover, MA). During the time period of study, the Ivenix Infusion System had a 510(k) (i.e., FDA premarket submission) pending and was not commercially available in the United States.

## Methods

This preliminary study was descriptive in design, with all pump programming conducted by the author. The author had never used any of the LVPs clinically and was not blinded to the different pump types. The primary outcome measure was the sequence

of keystrokes required to complete each programming task on each pump, presented as a detailed workflow mapping of each task on each pump. The workflows also included required keystrokes and all prompts and alerts encountered by the user during programming.

Before collecting any data, institutional review board approval was obtained from Hallmark Health System, where the author was employed at the time of study. All programming tasks were completed in a simulated setting by the author using the manufacturer recommendations for each pump and programming task.

## Study Procedures

The Alaris, Sigma Spectrum, and Plum A+ smart pumps were in acute care clinical use at the time of study. The Ivenix LVP was obtained from the manufacturer.

The following distinction was made between drug library and drug library profiles. The drug library was defined as the entire list of drugs available in the DERS. Each of the IV smart pumps in current



## The AAMI Foundation Supports Patient Safety

Our infusion therapy Quick Guides provide clear and practical guidance on:

- Smart pump compliance and drug libraries
- Optimizing patient outcomes
- Improving the use of multiple IV infusions

To download your free copy of these guides, visit [www.aami.org/foundation](http://www.aami.org/foundation)



clinical use contained the actual hospital drug library. The drug library profiles were defined as subsets of the total drug library, configured for the different clinical areas. For this study, only the currently available medical-surgical and critical care drug library profiles were used. Profiles make it easier for users to navigate the DERS by customizing the IV medications and fluids to those most commonly used by each clinical area. All programming for the study was completed using available medications and concentrations in either a medical-surgical or critical care drug library profile.

The software version for the pumps included in the study were as follows: Alaris, version 9.17; Sigma Spectrum, version 6.02.07; Plum A+, version 13.41.00.002; and Ivenix, version 1.1.0.10937.

All programming was completed using a single-channel setup and labeled IV fluids, medications, and tubing according to manufacturer recommendations. Each programming sequence began by powering on the device and programming a normal saline infusion at 125 mL/h using one of the available medical-surgical drug library profiles.

Task 1 involved administration of the secondary medication. The task was to administer cefazolin (1 g/50 mL) as a secondary infusion over 30 minutes using the medical-surgical drug library profile.

Task 2 involved administration of the weight-based infusion. The medication administered was dopamine (400 mg in 250 mL normal saline) delivered at 8 µg/kg/min as a continuous infusion. The patient weight was 70 kg.

## Data collection

The workflow mapping for each task on each IV smart pump was documented on an Excel spreadsheet during each programming task sequence. The programming time was defined as the time (in seconds) starting with the first keystroke and ending when the programming task was complete. The required programming time for each task was measured using the stopwatch feature on an iPhone and reported as an average of five programming attempts.

## Results

As previously described, before starting the workflow mapping for the secondary and weight-based infusion programming tasks, each pump was programmed to deliver normal saline at 125 mL/h from an available medical-surgical drug library profile. Even during this simple programming task, differences existed in the amount of time required for each system to power on, regardless of whether programming could be completed without the tubing loaded, and the number of keystrokes required for programming. Power on time was defined as the amount of time from the initiation of the “on” button until the system was ready for its first user interface interaction. These comparisons are shown in Table 1. A Pearson correlation between the number of required keystrokes and power on time was not significant (SPSS version 24; IBM, Armonk, NY).

The workflow mapping for secondary medication administration for each of the IV smart pumps is shown in Table 2. The

	IV Smart Pump Type (Manufacturer)			
	Alaris (Becton Dickinson)	Sigma Spectrum (Baxter)	Plum A+ (ICU Medical)	Ivenix Large-Volume Pump (Ivenix)
Power on time(s)	13.8	8.1	11.9	7.1
Number of required keystrokes for power on	17	15	10	16
Programming can be completed without tubing loaded	Yes	Yes	No	Yes

**Table 1.** Comparisons of power on time, number of keystrokes for power on, and need for tubing for programming for the intravenous (IV) smart pumps studied. Power on time was defined as the amount of time from the initiation of the “on” button until the system pump was ready for its first interaction. The power on times shown were calculated as an average of five attempts. Pearson correlation between time and number of programming steps was not significant.

workflow mapping for weight-based administration for each of the IV smart pumps is shown in Table 3. A summary of the data on the number of programming steps and programming time is shown in Table 4.

## Discussion

Differences in required workflow among the LVPs were easily noted from the initiation of the study. As highlighted in Table 1, differ-

ences were seen in power on time, ranging from 7.1 seconds for the Ivenix LVP to 13.8 seconds for the Alaris LVP. No correlation was found between the number of keystrokes required and the amount of time for power on, indicating that the number of required keystrokes may not be the most important factor informing power on time. The Ivenix prototype had the fastest power on time and was the only user interface with touchscreen

IV Smart Pump Type (Manufacturer)	Programming Step	Comments
Sigma Spectrum (Baxter)	1. Stop	—
	2. Review/program	—
	3. Program secondary	If you don't program the first letter quickly enough, the cursor moves (~1 s delay) and you have to use the back arrow.
	4. Type C and wait for cursor to move	—
	5. Type E, list comes up, use arrow to choose cefazolin (no need to hit "OK")	Multiple button pushes
	6. Use arrow to choose concentration	—
	7. OK	—
	8. Prompt: confirm concentration	—
	9. Yes	—
	10. Screen defaults to secondary (because primary already programmed). A large gray "2" appears in the background screen. Rate, VTBI, and time automatically displayed.	Multiple button pushes
	11. Must hit "OK," cannot go directly to RUN	—
	12. Prompt: secondary VTBI should equal secondary bag volume	—
	13. OK: not able to go directly to RUN	Not able to go directly to RUN
	14. Volume given automatically highlighted, hit "RUN"	—
	15. Prompt: Are drops falling in the secondary chamber and not the primary?	—
	16. Must hit "Yes," not able to go directly to RUN	—
Alaris (Becton Dickinson)	1. Channel select	On single-channel side module
	2. Secondary key	On PC (which is the main unit, sometimes referred to as the "brain")
	3. Select alphabet range that includes the letter C	Right side of PC screen
	4. Select the letter C	Right side of PC screen
	5. Select cefazolin	Left side of PC screen
	6. Select concentration	Left side of PC screen
	7. Is this correct? Select "Yes"	Right side of PC screen
	8. Review drug, volume, dose on PC screen, hit "Next"	—
	9. Verify that secondary clamp is open and hit "Start"	—

**Table 2.** Steps for programming a secondary infusion for each intravenous (IV) smart pump studied. Abbreviations used: PC, personal computer; VTBI, volume to be infused.

capabilities, thereby allowing users to navigate through required steps more quickly—a finding supported by previous research.<sup>9</sup> Three of the four LVPs in the study allowed programming to be completed without the tubing inserted, which is a usability issue with important clinical relevance. This feature can support clinical workflow by allowing pump preprogramming, which can be helpful to support clinical practice in cases where the medications needing to be programmed are known before the patient arrives, such as in cases of postoperative cardiac surgery. The Plum A+ was the only pump studied that did not have that capability.

The lack of similarities in the required programming steps for secondary medication infusions is evident in Table 2. The number of programming steps required ranged from nine with the Alaris pump to 16 with the Sigma Spectrum pump. In addition to requiring the highest number of programming

steps, the Sigma Spectrum had the most complex workflow, with many steps requiring multiple button pushes and the appearance of two nonactionable prompts (steps 12 and 15). Although use of nonactionable prompts is well intended and consistent with FDA recommendations for human factors medical device usability testing for regulatory approval,<sup>22</sup> users in daily clinical practice likely will become accustomed to these alerts and routinely bypass them without review. Thus, the presence of nonactionable alerts may add to complexity of IV smart pump programming with no safety benefit, as evidenced by the lessons learned from studies on alarm fatigue.<sup>23</sup> Findings from previous research support that the complexity of the Sigma Spectrum user interface for secondary medication administration resulted in longer programming times compared with other IV smart pumps.<sup>9</sup>

Similar to the findings for secondary medication infusion, the required program-

Table 2 continued

IV Smart Pump Type (Manufacturer)	Programming Step	Comments
Plum A+ (ICU Medical)	1. Select channel B	—
	2. Mode “piggyback,” select drug list	—
	3. Drug list page: arrow down to cefazolin	—
	4. Choices: standard, loading dose, or multistep; select “Standard”	—
	5. Enter drug dose	—
	6. Enter amount of fluid	—
	7. Enter dose	—
	8. Enter VTBI	Duration and rate calculated
	9. Message to review programming and confirm by hitting “Yes”	—
Ivenix Large-Volume Pump (Ivenix)	1. More options	—
	2. Program secondary	—
	3. Set fluid/drug	—
	4. Type CE	—
	5. Select CEFAzolin	—
	6. Select concentration	—
	7. Confirm	—
	8. Set duration	—
	9. 30 minutes	—
	10. Confirm	—
	11. Start secondary	—

ming workflow for weight-based infusion outlined in Table 3 shows no similarities among the different LVPs. The number of required steps ranged from 12 on the Plum A+ pump to 20 on the Ivenix pump. However, as shown in Table 4, no correlation was observed between the number of required steps and programming time. Although the number of steps is an important factor in programming time, it is not the only determinant of programming complexity.

Of important note, weight-based programming involves the highest-potency medications and is the most complex of all programming tasks. Anything that can be done to decrease weight-based programming complexity has the potential to result in fewer smart pump programming errors. Smart pump interoperability with the electronic health record (EHR), including both autoprogramming and -documentation, may be the most important next step in improving IV infusion safety. Although its use is currently limited, the number of hospitals with smart pump/EHR interoperability has increased from nine in 2013 to approximately 200 in 2017, with Alaris being the most used pump.<sup>24</sup>

Usability issues with potential clinical implications were noted on the Alaris IV smart pump during the workflow mapping. As seen in step 1, to change to a different drug library profile on the Alaris, the pump must be turned off and restarted. In the busy acute care setting, this is a cause for concern. A relevant example would be if a patient develops low blood pressure on the medical-surgical unit and requires emergency transfer to ICU for the initiation of vasopressor to support blood pressure. To change from the medical-surgical to critical care drug library profile on the same pump channel, the admitting ICU nurse would be required to power down/power on the pump—a process that takes approximately 22 seconds. With time as an essential factor, the ICU nurse is left with two undesirable choices: delay therapy or program outside of the DERS. Because programming outside of the DERS is an important factor in IV medication administration error, this technology limitation has the potential to

contribute to IV medication administration error. This usability requirement was apparent during the testing protocol described here and resulted in the Alaris having the longest programming time for weight-based infusions. None of the other IV smart pumps used in the study had this requirement.

Steps 7 and 8 highlight two additional Alaris usability issues with potential clinical implications. First, if a user inadvertently selects basic infusion instead of IV Guardrails during programming, no “back button” is available. To program within the DERS, the user must wait for the central PC unit to default back to channel select, which takes approximately 30 seconds. When this occurs, the user must decide whether to wait 30 seconds, potentially causing a disruption in workflow and/or a therapy delay, or bypass the DERS and use basic programming—a choice that would increase the likelihood of programming error.

Step 8 shows that when accessing the alphabet range to choose the desired medication, if the user pauses too long on any programming step, the pump defaults back to select channel, requiring the user to repeat all steps. Given that lack of time is one of the most common reasons for nurses to program outside of the DERS,<sup>25</sup> these usability issues may contribute to IV medication administration outside of the DERS and may have a relationship with IV medication infusion errors.

The relationships between the number of programming steps and the mean programming times for five attempts are shown in Table 4. No significant relationships were found between the number of programming steps and the time required for programming for either secondary medication administration or weight-based infusions. These findings are consistent with previous research.<sup>9</sup> The user interface is likely a more important factor in programming timing than the number of programming steps.<sup>9</sup> The Plum A+ pump was not included in these analyses because the author was not able to return to the clinical setting to complete the measurement of programming times before the end of the data collection period.

**Smart pump interoperability with the electronic health record, including both autoprogramming and -documentation, may be the most important next step in improving IV infusion safety.**

Smart Pump Type	Programming Step	Comments
Sigma Spectrum (Baxter)	1. Stop	—
	2. Clear program	Brings you back to the library profile list
	3. Use arrow to select critical care library	—
	4. OK	—
	5. Type first two letters of drug	Drug list comes up
	6. Use arrow to select drug	—
	7. OK	—
	8. Use arrow to choose concentration	—
	9. OK	—
	10. Confirm concentration, select "Yes"	—
	11. Enter weight in kilograms	—
	12. OK	—
	13. Enter dose	—
	14. OK	Autopopulates rate and VTBI
	15. OK	—
	16. RUN	—
	17. Answer "Yes" to prompt	Are all clamps open, is tubing kinked, and are drops flowing?
	18. Yes	Main screen showing both flow rate and dose display
Alaris (Becton Dickinson)	1. Must turn off pump in order to change to drug library profile from medical-surgical to critical care unit	—
	2. New patient? Select "No"	—
	3. Displays profile used last, select "No"	—
	4. Brings up drug library list, select critical care library	—
	5. Confirm	—
	6. Brings up list of available channels (only A), select channel	On side module (can use up to four channels on one PC)
	7. Select Guardrails drugs	On PC unit (if select basic infusion instead of IV guardrails, no back button—need to wait 30 seconds for unit to default back to channel select)
	8. Select alphabet range that includes the letter D	Right side of PC screen (if you pause too long on any programming step, goes back to select channel and must repeat all steps)
	9. Select the letter D	Right side of PC screen
	10. Select dopamine	Left side of PC screen
	11. Select concentration	Left side of PC screen
	12. Is this correct? Select "Yes"	Right side of screen
	13. Enter patient weight in kilograms	—
	14. Select "Next"	—
	15. Select rate or dose and the other will autopopulate	—
	16. Select VTBI	—
	17. Start	—

**Table 3.** Steps for programming a weight-based infusion on each intravenous (IV) smart pump studied. Abbreviations used: PC, personal computer; VTBI, volume to be infused.

Table 3 continued

Smart Pump Type	Programming Step	Comments
Plum A+ (ICU Medical)	1. While main channel running, choose options	—
	2. Change clinical care area	—
	3. Get drug library list and move to critical care	—
	4. Enter	—
	5. A	—
	6. Drug list	—
	7. Page down to dopamine	—
	8. Enter	—
	9. Standard program, concentration preloaded	—
	10. Enter weight in kilograms (dose, VTBI, duration, and rate autofill)	—
	11. Start	—
	12. Review and confirm, select "Yes"	—
Ivenix Large-Volume Pump (Ivenix)	1. Pause primary	—
	2. End primary	—
	3. Confirm end primary	—
	4. More options	—
	5. Program primary	—
	6. Set drug/fluid	—
	7. Touch medical-surgical profile display	—
	8. Brings up profile list, select critical care	—
	9. Type "DO"	—
	10. Select dopamine	—
	11. Confirm	—
	12. Select concentration	—
	13. Confirm	—
	14. OK	Administration alert for extravasation
	15. Set patient weight and dose rate	—
	16. Enter weight in kilograms	—
	17. Confirm	—
	18. Enter dose	—
	19. Confirm	—
	20. Start primary	—

	IV Smart Pump Type (Manufacturer)			
	Alaris (Becton Dickinson)	Sigma Spectrum (Baxter)	Ivenix Large-Volume Pump (Ivenix)	Plum A+ (ICU Medical)
<b>Secondary medication administration</b>				
Time(s)	15.3	24.5	13.6	NA
No. of steps	9	17	11	NA
<b>Weight-based infusion</b>				
Time(s)	48.6	20.6	20.3	NA
No. of steps	17	18	20	NA

**Table 4.** Summary of number of required steps and programming times for each intravenous (IV) smart pump studied. Programming times were calculated as an average of five attempts. No significant Pearson correlations were found between number of programming steps and programming time for either secondary or weight-based infusions. Abbreviation used: NA, not available.

**Because workflow complexity and programming times vary greatly among different IV smart pump types and programming tasks, studies designed to measure differences in the impact of interruptions for the most commonly used LVPs could help provide empirically derived improvements in IV smart pump medication administration safety.**

### Strengths and Limitations

The methodology used in this descriptive study is most closely aligned with comparative effectiveness,<sup>26</sup> making these findings potentially applicable to current clinical practice. The author endeavored to make the comparisons across the four LVPs used in this study as comparable as possible. To allow readers to assess the applicability of these findings to their own practice, devices in current clinical use were tested and the software versions were reported. However, with multiple configuration options available for all devices, and multiple versions of software in use at any given point in time, making ideal comparisons is difficult.

### Conclusion and Implications for Future Research

Although data support that the frequency of medication administration errors with IV smart pumps is still high,<sup>27</sup> little is known about what can and should be required of manufacturers to help make their products safer and easier to use. Because the current study was limited to manual programming steps, the impact of the need for bag height differential for secondary medication administration on programming complexity was not tested. However, of note, the Alaris and Sigma Spectrum IV smart pumps require a bag height differential for secondary medication administration. The Plum A+ and Ivenix IV pumps have technology that allows secondary medication administration which is independent of the primary and secondary bag heights. The patient safety implications of this difference require further research.

The impact of interruption on IV smart pump programming also requires additional research. Data support that interruptions during general medication administration contribute to medication error.<sup>28</sup> Two studies have found that interruption during patient-controlled analgesia pump programming contributes to medication error.<sup>8,29</sup> Although no published studies have measured interruption during medication administration using LVPs, it seems likely that the findings would be similar. Because workflow complexity and programming times vary greatly among different IV smart

pump types and programming tasks,<sup>9</sup> studies designed to measure differences in the impact of interruptions for the most commonly used LVPs could help provide empirically derived improvements in IV smart pump medication administration safety.

Because the vast majority of IV smart pump programming is still done manually, this study focused on usability from a manual use perspective. Even as autoprogramming and -documentation become more widely used, the ease of use for manual programming will continue to be relevant, as it will be required during downtime. Autoprogramming and -documentation are likely the next most important steps toward increasing IV infusion safety and thus require further research to elucidate their potential benefits to increased patient safety, as well as any unintended risks associated with their use.

### Funding

Funding for this study was provided by Ivenix. The study protocol, data collection, data analysis, and manuscript creation were independently completed by the author and without oversight from Ivenix.

### References

1. Husch M, Sullivan C, Rooney D, et al. Insights from the sharp end of intravenous medication errors: implications for infusion pump technology. *Qual Saf Health Care*. 2005;14(2):80–6.
2. Giuliano KK, Richards N, Kaye W. A new strategy for calculating medication infusion rates. *Crit Care Nurse*. 1993;13(6):77–82.
3. Ohashi K, Dalleur O, Dykes PC, Bates DW. Benefits and risks of using smart pumps to reduce medication error rates: a systematic review. *Drug Saf*. 2014;37(12):1011–20.
4. Hertz C, Sousa VD. The use of smart pumps for preventing medication errors. *J Infus Nurs*. 2009;32(5):257–67.
5. Murdoch LJ, Cameron VL. Smart infusion technology: a minimum safety standard for intensive care? *Br J Nurs*. 2008;17(10):630–6.
6. Rothschild JM, Keohane CA, Cook EF, et al. A controlled trial of smart infusion pumps to improve medication safety in critically ill patients. *Crit Care Med*. 2005;33(3):533–40.

7. Nuckols TK, Bower AG, Paddock SM, et al. Programmable infusion pumps in ICUs: an analysis of corresponding adverse drug events. *J Gen Intern Med.* 2008;23(suppl 1):41–5.
8. Campoe KR, Giuliano KK. Impact of Frequent Interruption on Nurses' Patient-Controlled Analgesia Programming Performance. *Hum Factors.* 2017;59(8):1204–13.
9. Giuliano KK. IV Smart Pumps: The Impact of a Simplified User Interface on Clinical Use. *Biomed Instrum Technol.* 2015;fall suppl:13–21.
10. Alexander G, Staggers N. A systematic review on the designs of clinical technology: findings and recommendations for future research. *ANS Adv Nurs Sci.* 2009;32(3):252–79.
11. Rajkomar A, Blandford A. Understanding infusion administration in the ICU through Distributed Cognition. *J Biomed Inform.* 2012;45(3):580–90.
12. Namshirin P, Ibey A, Lamsdale A. Applying a multidisciplinary approach to the selection, evaluation, and acquisition of smart infusion pumps. *J Med Biol Eng.* 2011;31(2):93–8.
13. Ali H, Li H, Lisboa P. Usability analysis and redesign of infusion pump user interface. Paper presented at the International Symposium on Human Factors and Ergonomics in Health Care, Baltimore, MD, April 26–29, 2015. Santa Monica, CA: Human Factors and Ergonomics Society.
14. Schmettow M, Vos W, Schraagen JM. With how many users should you test a medical infusion pump? Sampling strategies for usability tests on high-risk systems. *J Biomed Inform.* 2013;46(4):626–41.
15. Schraagen JM, Verhoeven F. Methods for studying medical device technology and practitioner cognition: The case of user-interface issues with infusion pumps. *J Biomed Inform.* 2013;46(1):181–95.
16. Maddox RR, Danello S, Williams CK, Fields M. In: Henriksen K, Battles JB, Keyes MA, Grady ML, eds. *Advances in Patient Safety: New Directions and Alternative Approaches*. Vol. 4: Technology and Medication Safety. Rockville, MD: Agency for Healthcare Research and Quality; 2008:1–14.
17. Huber C, Rebold B, Wallace C. *ERCI Institute PSO Deep Dive Analyzes Medication Events*. Plymouth Meeting, PA: ECRI Institute; 2012.
18. Cassano-Piché A, Fan M, Sabovitch S, et al. Multiple intravenous infusions phase 1b: practice and training scan. *Ont Health Technol Assess Ser.* 2012;12(16):1–132.
19. Easty T, Cassano-Piché A, White R. *Multiple Intravenous Infusions Phase 1a: Situation Scan Summary Report*. Toronto, Canada: Health Technology Safety Research Team, University Health Network; 2010.
20. Herout PM, Erstad BL. Medication errors involving continuously infused medications in a surgical intensive care unit. *Crit Care Med.* 2004;32(2):428–32.
21. IHS Markit. *Infusion Pumps and Accessories: World*. Englewood, CO: IHS Markit; 2013.
22. Food and Drug Administration. *Applying Human Factors and Usability Engineering to Medical Devices: Guidance for Industry and Food and Drug Administration Staff*. Available at: [www.fda.gov/downloads/MedicalDevices/.../UCM259760.pdf](http://www.fda.gov/downloads/MedicalDevices/.../UCM259760.pdf). Accessed March 13, 2018.
23. Ruppel H, Funk M, Whittmore R. Measurement of Physiological Monitor Alarm Accuracy and Clinical Relevance in Intensive Care Units. *Am J Crit Care.* 2018;27(1):11–21.
24. Hess P, Warburton P. *Smart Pump/EMR Interoperability*. Orem, UT: Klas; 2017.
25. Giuliano KK, Ruppel H. Are smart pumps smart enough? *Nursing.* 2017;47(3):64–6.
26. Giuliano KK, Ferguson M, Silfen E. Medical technology innovation and the importance of comparative effectiveness research. *Journal of Medical Marketing.* 2012;12(1):55–66.
27. Schnock KO, Dykes PC, Albert J, et al. The frequency of intravenous medication administration errors related to smart infusion pumps: a multihospital observational study. *BMJ Qual Saf.* 2017;26(2):131–40.
28. Westbrook JI, Coiera E, Dunsmuir WT, et al. The impact of interruptions on clinical task completion. *Qual Saf Health Care.* 2010;19(4):284–9.
29. Hicks RW, Sikirica V, Nelson W, et al. Medication errors involving patient-controlled analgesia. *Am J Health Syst Pharm.* 2008;65(5):429–40.