chi+med
making medical devices safer

EPSRC Programme Grant EP/G059063/1

Public Paper no. 20

Comparing Actual Practice and User Manuals: A Case Study based on Programmable Infusion Pumps

Ann Blandford, Abigail Cauchi, Paul Curzon, Parisa Eslambolchilar, Dominin Furniss, Andy Gimblett, Huayi Huang, Paul Lee, Yunqiu Li, Paolo Masci, Patrick Oladimeji, Atish Rajkomar, Rimvydas Ruksenas & Harold Thimbleby

Proceedings of EICS4Med, First International Workshop on Engineering Interactive Computing Systems for Medicine and Health Care

PP release date: 25 May 2011

file: WP020.pdf

Comparing Actual Practice and User Manuals: A Case Study Based on Programmable Infusion Pumps

Ann Blandford[4] Abigail Cauchi[3] Paul Curzon[1] Parisa Eslambolchilar[3] Dominic Furniss[4]
Andy Gimblett[3] Huayi Huang[1] Paul Lee[2] Yunqiu Li[3] Paolo Masci*[1] Patrick Oladimeji[3]
Atish Rajkomar[4] Rimvydas Rukšėnas[1] Harold Thimbleby[3]

(1) Queen Mary University of London, (2) Singleton Hospital, Swansea,
(3) Swansea University, (4) University College London

CHI-MED: Computer-Human Interaction for Medical Devices
www.chi-med.ac.uk

ABSTRACT

We report on a case study investigating current practice in the use of a programmable infusion pump. We start by formalising an existing description of the procedure followed by nurses for setting up a commercial infusion pump obtained via observation and interview. We then compare and contrast this procedure with a formal description of the sequence of actions reported in the pump's user manual. Mismatches were validated by a training manager. The aim of this comparison is to point out how minor mismatches between the two descriptions can be used to reveal major safety issues. Our contributions are: first, we analyse a real-world system and show the importance of having a clear and consistent specification of the procedures; second, we show how a graph-based notation can be conveniently used as the basis for building non-ambiguous and intuitive specifications in higher-order logic. We argue that this can provide support to an investigator when building a description of actual practice in that it can help focus attention on areas to observe more closely and on questions to ask to understand why procedures, as followed, are the way they are.

Keywords

Medical devices; Formal methods.

1. INTRODUCTION AND MOTIVATION

The use of medical devices such as infusion pumps is safety-critical and therefore it is vital that safe procedures are followed. Manufacturers set out such procedures in the manuals. However, with any complex technology there is often a mismatch between the prescribed usage and the actual behaviour of people using it. Such behaviour is not necessarily unsafe or even wrong. It can occur due to differences in context from the expected, or workarounds to improve efficiency for example.

Manuals can be used as one of the approaches for designers to express their conceptual model. Furthermore, manuals can be one of the approaches used by designers to communicate their conceptual model of the system to users. However, when device users are developing their mental models of the same system, the manual only fills part of the picture. The procedure recommended by the designer is first interpreted by the training staff, before it reaches the nurses. The nurses then construct their mental models of how the system should work through training and interaction before using it in practice. In both stages, the model is influenced by other factors, such as existing conventions used to program other devices, personal preferences and sometimes the capability of interpreting technical terms.

However, if there is such a mismatch between what nurses are *trained to do*, what the manual says or what happens in reality, then this mismatch might be dangerous. It could be for example, that those who developed the workarounds were not aware of the reasons for the prescribed procedure and so are omitting safety checks [6]. It is therefore important to have a clear, rigorous and consistent way to analyse the procedures prescribed in training, those in manuals and actual practice in fine detail. This would allow us to better understand these procedures and so detect potentially dangerous mismatches to ensure that we do not facilitate the widespread acceptance of these potential "latent errors".

We use a graph-based notation for building an abstract description of current practice and user manuals, and we translate such a description into higher-order logic for generating a non-ambiguous specification suitable to be analysed in automated reasoning tools, such as PVS [7] or SAL [2]. In order to trial our ideas, here we consider a case study on programmable infusion pumps: we show how actual practice and user manuals can be specified in higher-order logic,

*Corresponding author.

Figure 1: B.Braun Infusomat Space

and we present the insights we gained on the design of procedures by analysing such a formal specification.

The paper is organised as follows. In Section 2, we report on related work on formal methods applied to the analysis of procedures and user manuals. In Section 3, we overview the procedure followed by nurses for setting up an infusion pump. In Section 4, we describe how we translated informal descriptions into a higher-order logic specifications. In Section 5, we present the analysis carried out and the obtained results. In Section 6, we conclude the paper.

2. RELATED WORK

Several projects have explored how to combine formal modelling with user models as a way to obtain accurate user manuals. For example, [9, 11] show how a user manual can be automatically generated from a logic specification of a design and that such an automated process can help detect errors in the design.

Massink et al. [4] explore how model checking can be used to generate traces from a specification to answer how to and what if questions posed in temporal logic. Weitl et al [12] explore how to maintain consistency of context dependent documents including training manuals, focusing on user support by a pattern-based specification methodology. They combine temporal logic, ontologies, and a pattern-based specification approach.

Hebert [3] investigated the degree to which manuals were user tested to check their quality in six high-tech companies through interviews with key people. Where testing was done, testing methods were generally found to be limited.

3. PROCEDURE FOR SETTING UP A PROGRAMMABLE INFUSION PUMP

In this section, we overview the description of the procedure followed by medical practitioners for setting up programmable infusion pumps as analysed in the study [8] that the work described here is based on. For the purposes of this paper, we focus on the sequence of actions carried out by nurses for starting a constant flow rate infusion. For clarity of exposition, we use a slightly re-worded version of the original description here.

In the particular hospital studied, the B.Braun Infusomat Space infusion pump was used (see Figure 1). The pump has up, down, left and right arrow buttons for navigating through menus and entering numbers. The dot matrix display of the pump shows the current state of the pump as well as instructions of what the buttons do in that current state to aid user interaction.

3.1 Setting up a Constant Flow Rate Infusion

In intensive care units, nurses prepare and administer drugs according to prescriptions decided by doctors. Specifically, nurses prepare the medication, fill a bag with it, and then administer it through an infusion pump. The informal description of the activities carried out by nurses follows. This description describes the process in a single hospital as obtained through observation and interviews. In the description here, we will use a level of detail appropriate for the purposes of this article only. Readers interested in the full, original description, which has been validated by a senior nurse, should refer to [8].

Preparing a bag with the medication. In the particular hospital studied, the list of drugs to be administered to a patient and the details for preparing the medication are reported in the electronic patient record. Nurses read such information and proceed as follows: they prepare the bag with the medication and stick a descriptive label onto the bag; they connect the infusion line to the bag; they prime the infusion line, i.e., they push the fluid to the tip of the infusion line in order to bleed all air bubbles from the infusion line.

Starting the infusion. The infusion pump is turned on by pressing the On/Off button. When the pump is turned on, the nurse can open the pump door by pressing the Open button, and insert the infusion line in the pump: the infusion line end connected to the bag must be placed into the Inlet Line, while the other end must be fit into the Outlet line. The pump door can be closed after inserting the line in the pump. At this point, the pump displays the main menu, and the nurse can set up the infusion rate by selecting the Infusion Rate menu item. This item is the first in the main menu of the pump, and is highlighted by default. When setting up the infusion rate, the pump displays a number of digit spots with a decimal point ("_ _ _ . _ _"), and the first digit spot is selected by default. The nurse can use the Up/Down Arrow buttons for increasing/decreasing the digit, and the Left/Right Arrow buttons for highlighting a digit. Once the correct digits have been entered, the nurse presses the Ok button to confirm the entered value. Once the value has been confirmed, the pump displays the main menu. At this point, the nurse presses the Start/Stop button to start the infusion.

4. FORMAL SPECIFICATION

In this section, we translate the informal description of the activities carried out by nurses into a formal specification. Such a specification can be mechanically analysed with automated reasoning tools, such as in [10]. The notation we use draws concepts from Activity Networks [5], a widely used formalism for modelling complex concurrent systems, and relies on higher-order logic to formally specify both the dependency relations among activities and the activities themselves.

The formal specification for the activities carried out by the

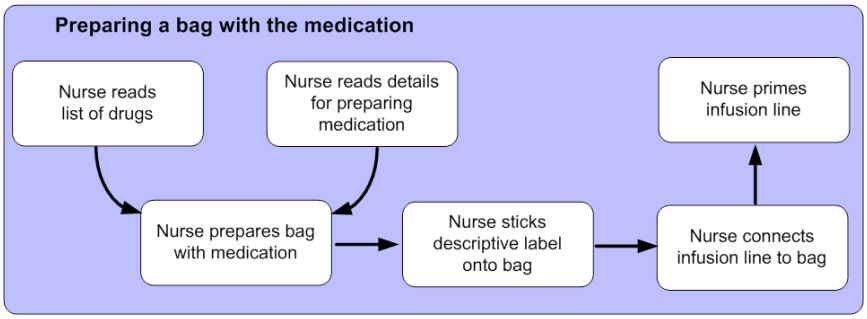

Figure 2: Procedure followed by nurses for preparing a bag with the medication

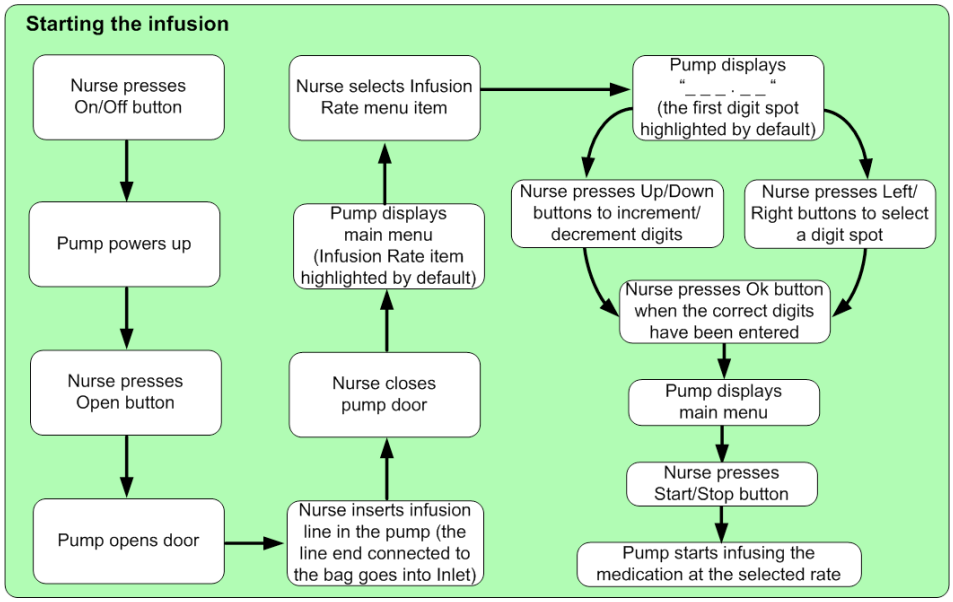

Figure 3: Procedure followed by nurses for starting the infusion

nurse in preparing a bag with the medication and for starting the infusion is graphically depicted as a labelled graph in Figures 2 and 3. In the graph, labelled nodes represent activities, and edges between nodes represent causal and temporal dependency relations among activities. An activity can be performed only if all directly connected activities have already been performed. For instance, consider the activity *"Nurse prepares bag with medication"* shown in Figure 2; such an activity can be performed only if two other activities have already been performed (*"Nurse reads list of drugs"* and *"Nurse reads details for preparing medication"*). Edges in the graph may have labels that specify control-flow conditions.

In the following, we show how activities can be specified in higher-order logic. To this end, we describe an excerpt from our formal specification developed in the PVS specification and verification system [7]. The PVS specification language builds on classical typed higher-order logic with the usual base types (e.g., `bool`, `nat`, `integer` and `real`), function type constructors `[A -> B]` (predicates are functions with boolean range type), structured data types, and abstract data types. PVS specifications are packaged as theories. PVS theories can use definitions and theorems of other theories by importing them. Providing a detailed description of the developed PVS theories is beyond the scope of this paper.

In the following we outline the formal definition of a simple activity from the theory concerned with preparing a bag. Our aim is to give a feel for the style of specification used. In the `prepare_bag_th` theory, given in Figure 4, we have two type definitions: the first definition specifies the pump state as a structured data type (`pump_state`) containing two fields (`display`, of type `message`, and `door`, of a user-defined enumerated type); similarly, the other type definitions specify the bag state as an enumerated data type, and the system state as a structured data type with four fields, which represent the pump, the bag, the infusion line, and the roller clamp.

The activities carried out by nurses are specified as transition functions over the system state. In our example, in Figure 4, we specify the activity *"Pump powers up"* as a (higher-order) function that changes the state of the pump according to the pressed button, which is specified as function parameter. The keyword `LAMBDA` here just specifies that a function follows, taking in this case a single argument —

```
prepare_bag_th: THEORY BEGIN %-- imports omitted
  pump_state  : TYPE =
    [# display: message,
       door   : bool #] %-- 1=open, 0=closed
  bag_state   : TYPE =
    {new_bag, labelled, empty}
  system_state: TYPE =
    [# pump: pump_state,
       bag: bag_state,
       roller_clamp: bool #] %-- 1=open, 0=closed
  pump_powers_up(b: button):
   [system_state -> system_state] =
    LAMBDA(sys: system_state):
      IF b = On_Off_button
       THEN LET new_pump_state = pump(sys)
            WITH [ display := turned_on ]
            IN sys WITH [ pump := new_pump_state ]
      ELSE sys ENDIF
  %-- more definitions omitted
 END prepare_bag_th
```

Figure 4: The formal specification of an example activity (pump powers up)

the system state — and returning a new system state as determined by an `IF` statement. If the button pressed is the On/Off button, then the new pump state part of the system state is the same as the old one but with the display part of the state turned on.

The activity described above is just one activity. We also formalised all other activities depicted in Figure 4. The advantage of providing a formal specification such as this is that it gives a very precise and unambiguous description of the procedure that is not open to different interpretation as with an informal description. This is because the language used has a precise and well defined mathematical meaning. This also opens up the possibility of using mathematical analysis tools to explore the consequences of the procedure and compare it with others.

5. ANALYSIS

We carried out two kinds of analysis on the formal specification of the procedure. The first analysis was intended to ensure semantic constraints, such as checking the consistency of control flow conditions, and the completeness of the activities' specification. This kind of analysis has the potential to detect inconsistencies between different parts of the specification and that nothing is missing from the behaviour described. For example it could highlight decision points where the consequences of only one branch of the decision has been specified. The second kind of analysis compares the procedures carried out by nurses in the real system (i.e., the best practice) against the sequence of actions specified in the user manual, and aims at pointing out possible mismatches that may lead to safety issues. This is done by creating two formal descriptions of the procedures, one from the observed behaviour and one based on what is described in the manual.

5.1 Checking Semantic Constraints

While translating the informal description of the procedure into a higher-order logic specification, we discovered two issues. The first issue is due to the inherently ambiguous semantics of natural language, and affects the description of activity *"Pump displays "_ _ _ . _ _" (the first digit spot is highlighted by default)"*. The activity is under-specified, since it is not clear which is the first digit spot. Is it the left-most spot? Is it the first integer spot? We checked the real pump, and the right interpretation is the latter one. The second issue is related to numeric entries. The description reported in Figure 3, indeed, reports that *"Nurse presses Up/Down arrow buttons to increment/decrement digits"*, while this is not the case in the real pump –for instance, the up button, which is used to increase the infusion rate, is also used as a *recall memory* button in certain device modes.

5.2 Actual Practice and the User Manual

In comparing the actual practice with that described in the B.Braun user manual a range of significant differences emerged.

The procedure for preparing a bag as described in the manual is relatively simple consisting of only four steps (see Figure 5). If we compare this description with the actual procedure followed by nurses (which is reported in Figure 2), we can notice two main differences.

1. In the actual procedure, nurses prime the infusion line, while the manual reports only that the infusion line must be filled from bottom to top, and postpones priming at a later stage, just before establishing the connection with the patient (see Figure 6).

2. The manual explicitly reports that the bag must not be below the pump level (see the grey box of Figure 5); this detail is omitted in the description of the actual procedure.

We have checked the procedure with a medical devices training manager. The procedure actually followed by nurses is correct: nurses should not simply fill the infusion line, but also prime the line. Also, nurses are taught to place the bag above the pump level: the omitted detail is probably due to the physical setup in the hospital observed, which constrains the bag to be always above the pump level (the bag needs to be attached to a hook which is fixed above the pump level). This effectively removes the need for the step of checking that the bag is not below the pump level, and could be why nurses did not report this step when describing the procedure they follow.

Several mismatches are highlighted in the procedure for starting the infusion with respect to the actual procedure followed by nurses. Two main types of mismatches can be evidenced:

Missing details on constraints and precautions. The manual provides a number of constraints that should be ensured by nurses while starting the infusion (e.g., nurses must ensure that the pump is properly installed before turning the pump on), and precautions during particular activities (e.g., nurses must never leave the pump unattended while inserting the infusion line).

Different sequence of actions. The manual explicitly reports that the pump can be used to prime the infusion line, but also states that such function can be disabled. When the function is disabled, the manual simply proceeds to the

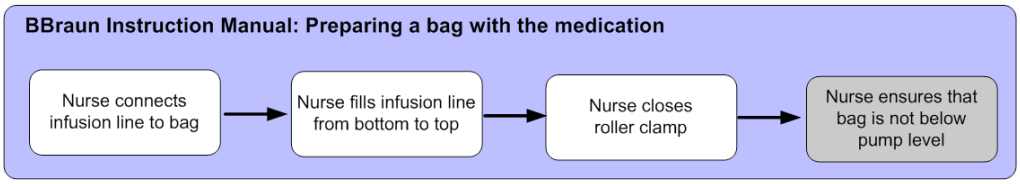

Figure 5: Procedure specified in B.Braun's Manual for preparing a bag with the medication

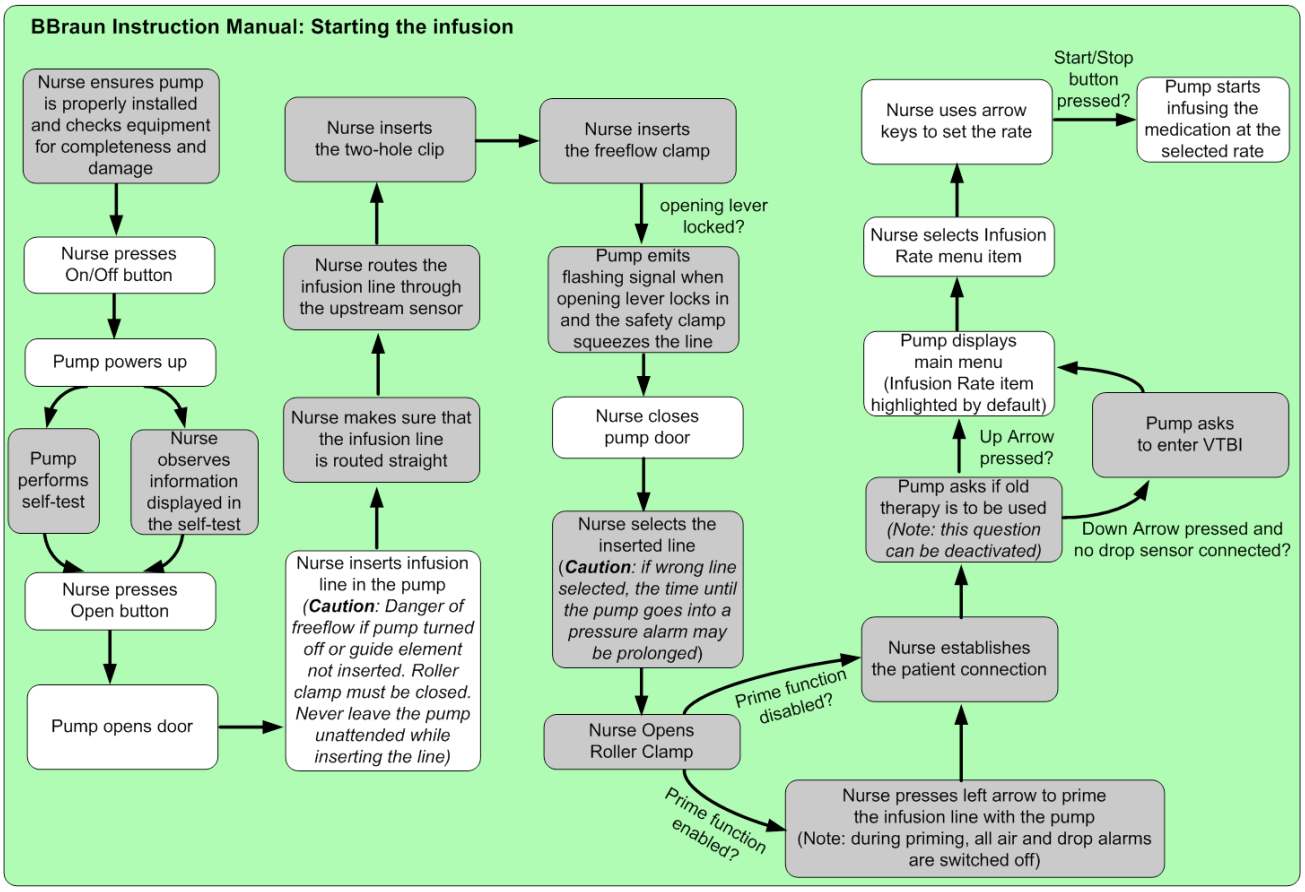

Figure 6: Procedure specified in B.Braun's Manual for starting the infusion

next step ("*Press the Up Arrow if the prime function is enabled. [...] Then press Down Arrow to proceed. Establish the patient connection*" [1]).

We engaged a medical devices training manager for discussing the above differences. It turns out that the missing details on constraints and precautions are not only related to activities that nurses simply omit to explicitly state (e.g., nurses are aware of the situations in which there is a danger of free-flow), but there are also a number of activities that nurses are actually not able to perform due to the particular work context, which is always very busy and does not allow nurses to dedicate time to certain activities, such as observing the messages shown by the pump during the self-test.

Regarding the different sequence of actions, the most critical difference is related to priming the infusion line. The manual reports that it can be done with the pump, given that the prime function is enabled on the pump; if the function is disabled, the next step is to establish the patient connection (see Figure 6). However, before starting the infusion, the line must be primed, otherwise there is a safety issue for the patient. The manual omits to point out this detail, and the procedure, as described in the manual, is not best practice. The medical devices training manager reported that, in fact, there were incidents due to nurses and trainers following procedures reported in the manuals, like the one described.

6. DISCUSSION AND CONCLUSIONS

We have described a study that compares a formal specification of actual practice for setting up an infusion pump with a formal specification of the sequence of actions described in the user manual of the pump. The study is based on a real pump and on observations made in a real hospital.

We discovered several issues and mismatches, such as ambiguity in the language used in the descriptions, and various checks prescribed in the manual of the pump but omitted in the actual practice. Our study shows how a model-based engineering approach can be conveniently used to support an investigator to understand workflow situations. Such issues are easily missed when using only informal descriptions.

The formalisation process uncovered an issue in the user manual related to priming the line (the procedure described in the manual was not consistent with the actual best practice). This demonstrates that a model-based engineering approach could be used to improve manuals and training material. Whilst this study concerned a B.Braun pump we believe that similar issues would apply with the pumps of other manufacturers and their actual use.

Understanding how devices are used in practice and pointing out issues with manuals is of potential use to manufacturers, trainers and procurement staff. For example, if staff are using workarounds then it is important for device designers to be aware of this so that they can determine whether these practices are safe.

Providing a direct support in training literature is likely to provide a commercial advantage for manufacturers: if such workarounds are adapted to context to make the pump use easier or more effective, then the pump in question would better support the nurses in doing their work. On the other hand if the practices are unsafe, it is in the manufacturer's interest to design the device in ways to remove the need or possibility of such workarounds. Similarly, if a device design is such that unsafe practices result to deal with the realities of the job, then training and procurement staff need to know how to improve procurement decisions and training regimes. Also, if training staff are aware of why procedures are the way they are, then they can help spread good practice. Formal specifications can be simulated, and this opens up the possibility of automatically generating visualisation tools of scenarios for use in training (either to show bad consequences of actions or to illustrate ways to improve performance and safety).

Whilst in this study the investigation of actual behaviour was complete when the formal work was carried out, these techniques could also help an investigator collecting ethnographical data and actually build a deeper understanding of the system as observations and interviews take place. For example, insights gained during model-based analysis can help focus attention on areas to observe more closely and questions to ask. When only an informal description is being used, such issues can easily be missed.

Current tools are not at a stage of maturity that an actual investigator could easily develop a formal specification and analyse it, though this may be possible in the future as the techniques are developed and built into tools. In the meantime, however, the approach can be applied if a formal methods expert is paired with the investigator to do model development and analysis and interpret the results in dialogue with the investigator.

Acknowledgements

Funded as part of the CHI+MED: Multidisciplinary Computer-Human Interaction research for the design and safe use of interactive medical devices project, EPSRC Grant Number EP/G059063/1, Formally-based tools for user interface analysis and design, EPSRC Grant Number EP/F020031/1, and Extreme Reasoning Grant Number EP/F02309X/1.

7. REFERENCES

[1] B|Braun Melsungen AG. Infusomat space and accessory: Instruction for use (ver. 686e).

[2] Leonardo de Moura, Sam Owre, Harald Ruess, John Rushby, N. Shankar, Maria Sorea, and Ashish Tiwari. SAL 2. In Rajeev Alur and Doron A. Peled, editors, *Computer Aided Verification: CAV 2004*, volume 3114 of *Lecture Notes in Computer Science*, pages 496–500. Springer-Verlag, July 2004.

[3] Susan M. Hebert. The cat in the reaper: Usability testing of software instruction manuals. Master's thesis, San Jose State University, 1989.

[4] M. Massink and D. Latella. *Deriving Manuals from Formal Specifications*, volume 1. 2003.

[5] A. Movaghar and J.F. Meyer. Performability modelling with stochastic activity networks. In *Proceedings of the 1984 Real-Time Systems Symposium*, pages 215–224, 1984.

[6] Donald A. Norman. *The Design of Everyday Things*. Basic Books, New York, reprint paperback edition, 2002.

[7] S. Owre, S. Rajan, J.M. Rushby, N. Shankar, and M.K. Srivas. PVS: combining specification, proof checking, and model checking. In Rajeev Alur and Thomas A. Henzinger, editors, *Computer-Aided Verification, CAV '96*, number 1102 in Lecture Notes in Computer Science, pages 411–414, New Brunswick, NJ, July/August 1996. Springer-Verlag.

[8] Atish Rajkomar. Extending distributed cognition analysis for complex work settings: a case study of infusion pumps in the intensive care unit., 2010. MSc Thesis.

[9] H. Thimbleby and P. B. Ladkin. From logic to manuals. *Software Engineering Journal*, 11(6):347–354, 1997.

[10] Harold Thimbleby. Contributing to safety and due diligence in safety-critical interactive systems development by generating and analyzing finite state models. In *Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems*, EICS '09, pages 221–230, New York, NY, USA, 2009. ACM.

[11] Harold W. Thimbleby and Peter B. Ladkin. From logic to manuals again. *IEEE Proceedings - Software*, 144(3):185–192, 1997.

[12] Franz Weitl, Mirjana Jakšić, and Burkhard Freitag. Towards the automated verification of semi-structured documents. *Data Knowl. Eng.*, 68:292–317, March 2009.