Int. J. Human-Computer Studies 72 (2014) 284–297

Contents lists available at ScienceDirect

Int. J. Human-Computer Studies

journal homepage: www.elsevier.com/locate/ijhcs

Analysing interactive devices based on information resource constraints☆

José Creissac Campos [a,b], Gavin Doherty [c], Michael D. Harrison [d,e,*]

[a] Departamento de Informática, Universidade do Minho, Braga , Portugal
[b] HASLab/INESC TEC, Braga, Portugal
[c] Lero@TCD, School of Computer Science and Statistics, Trinity College Dublin, Ireland
[d] School of Electrical Engineering and Computer Science, Queen Mary University of London, London, United Kingdom
[e] School of Computing Science, Newcastle University, Newcastle upon Tyne, United Kingdom

A R T I C L E I N F O

Article history:
Received 13 February 2013
Received in revised form
1 August 2013
Accepted 2 October 2013
Communicated by Fabio Paterno
Available online 28 October 2013

Keywords:
Formal analysis
Task analysis
Distributed cognition
IV infusion pumps

A B S T R A C T

Analysis of the usability of an interactive system requires both an understanding of how the system is to be used and a means of assessing the system against that understanding. Such analytic assessments are particularly important in safety-critical systems as latent vulnerabilities may exist which have negative consequences only in certain circumstances. Many existing approaches to assessment use tasks or scenarios to provide explicit representation of their understanding of use. These normative user behaviours have the advantage that they clarify assumptions about how the system will be used but have the disadvantage that they may exclude many plausible deviations from these norms. Assessments of how a design fails to support these user behaviours can be a matter of judgement based on individual experience rather than evidence. We present a systematic formal method for analysing interactive systems that is based on constraints rather than prescribed behaviour. These constraints capture precise assumptions about what information resources are used to perform action. These resources may either reside in the system itself or be external to the system. The approach is applied to two different medical device designs, comparing two infusion pumps currently in common use in hospitals. Comparison of the two devices is based on these resource assumptions to assess consistency of interaction within the design of each device.

1. Introduction

The process of assessing the usability of an interactive system is often criticised, either because it is biased by the expertise and judgement of the analyst or because it requires a sufficiently developed system to be able to assess it through user trials, either in the laboratory or in the "wild". These issues are particularly important when the system is safety or business critical. In such circumstances evaluation could make it necessary to make significant and potentially expensive changes late in the development process. User testing, while valuable, is also unlikely to cover all plausible user interactions with the system.

To explore usability, one starting point is to consider behaviours that achieve the intended goals of an activity (Butterworth et al., 1998). The focus of concern must be with what people might do with a device. This concern can be contrasted with a more complete analysis of every behaviour that an interactive device is capable of. Many of these behaviours, though undesirable, are unlikely to be carried out by a real user. Focusing on more likely user behaviour is often done by considering tasks or scenarios because they provide typical or intended behaviours. The problem is that what the designer intended or the scenario envisaged is not always how the system is actually used. Unexpected uses of the device can lead to entirely unforeseen usability issues. Our approach is based on a more situated view of interaction. It is assumed that interaction is shaped moment-by-moment by the information and affordances provided by the device, or the environment of the device. A typical interaction design issue that illustrates the role of resources is the "keyhole problem" in which users are distracted from achieving their primary goals by, for example, accessing different screens within a hierarchical menu structure in order to gather information that they require (Woods et al., 1994). The cues that are required to maintain an awareness

☆This paper has been recommended for acceptance by Fabio Paterno.

* Corresponding author at: School of Computing Science, Newcastle University, Newcastle upon Tyne, NE1 7RU UK.
Fax: +44 1912228232.
E-mail addresses: jose.campos@di.uminho.pt (J.C. Campos), Gavin.Doherty@tcd.ie (G. Doherty), michael.harrison@eecs.qmul.ac.uk, michael.harrison@ncl.ac.uk (M.D. Harrison).

1071-5819/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.ijhcs.2013.10.005

of the main goal and their progress towards it are lost as the user navigates through the information space, viewing only a small proportion of the available information at a time. This places unreasonable demands on short-term memory and introduces vulnerability to a range of errors. Thus information needs provide constraints on user behaviour, and together with the resources afforded by the system, act to shape the likely behaviours of the end user. Users are therefore considered to behave by following paths that are suggested or enabled by *information resources*. Using them to drive analysis enables consideration of a broader class of uses and user behaviours than could be achieved by restricting analysis to the behaviours encoded in more prescriptive models. This makes it possible to explore many more behaviours than would be represented by, say, a task model (Kirwan and Ainsworth, 1992; Mori et al., 2002), and in some cases to realise that what the user might do is not consistent with what the designer had intended.

The proposed method involves specifying assumptions about information resources so that it is possible to check whether the right information is provided to users at the right time in support of activities. The designer must consider a range of issues when deciding which information is relevant at a given stage in the interaction: support for a range of user strategies, making the most of available screen space, avoiding information overload, and reconciling competing information requirements when the system supports a number of different activities. Subsequently, a model checker is used to find the more plausible behaviours within the space of all possible interactions. Traces that are generated by the analysis contain actions that are constrained by information resource assumptions. The model checking tool simplifies the process by automating the generation of these constrained behaviours. Each trace represents a scenario which is plausible with respect to the information resources and possibilities for action available to the user. Once these scenarios have been generated they can be explored by domain, software engineering, and human–computer interaction experts to consider their implications for design, and to decide whether remedial action needs to be taken (such as changing the device design). Some traces might be unexpected, perhaps bypassing some step which is important from a safety perspective, they might be longer than expected (as more efficient paths are insufficiently resourced), or there might not exist any well-resourced path to achieving the user's goal. The advantage of this approach is that it allows the analyst to focus on a subset of possible scenarios, ignoring implausible behaviours. It is always possible to model check the device without ignoring implausible interactions but this is likely to generate too many uninteresting behaviours, rendering this interdisciplinary analysis impractical.

This technique is designed to complement the systematic analysis of interactive devices using batteries of properties proposed by Campos and Harrison (2008, 2009) and Harrison et al. (2013). In the case of the systematic analysis no assumptions about use are made except insofar as they are captured in the properties themselves. For example, a property might state that a certain confirmation action is always performed, unless the user cancels the interaction. The property says nothing about whether the actions are sufficiently salient to be easy to use for example.

The paper extends work published by Campos and Doherty (2006) and Doherty et al. (2008). It compares two real systems that were both developed to support IV (intravenous) infusion in a hospital context. While the technique is intended to be generic to a range of modelling approaches it is illustrated using the IVY tool. This tool was initially developed to support the systematic analysis of interactive systems. The paper demonstrates a scaleable method for analysing interactive systems using constraints based on information resources, and to demonstrate the analysis of consistency properties and comparisons between different devices.

More specifically, the paper's contributions are:

- It demonstrates the use of *resources* as a modelling concept.
- It shows how resources can be used to focus analysis on plausible sequences.
- It illustrates the technique by contrasting the resources required in one real-world example with those in another both designed to support the same activities.
- The method also captures a number of different precise notions of task consistency and applies them.

The paper first discusses the background to this resource based approach (Section 2). Section 3 explains how resources can be used to support the analysis of systems in use. The manner in which resources are specified and the way in which goals are used in property formulation are discussed, along with the possibilities for tool support. This section introduces the proposed method. Models of the two infusion pumps are then briefly introduced in Section 4. Section 5 describes the activity context for describing the two devices before providing a discussion of the resource constraints relevant to the example (Section 6). The penultimate section uses the model of activities and description of the resources to compare the two models of devices (Section 7). Finally, discussion of the wider application of the method and of further developments is to be found in Section 8.

2. Background

The use of behavioural models, focusing on the system and supported by automated reasoning, to analyse human–computer interaction has been the subject of previous research (Mori et al., 2002; Campos and Harrison, 2001; Rushby, 2002; Loer, 2003). The particular tool that underpins the analysis performed in this paper is *model checking* (Clarke et al., 1999). Model checking is an automated approach that verifies that a formal model of a system satisfies a set of desired properties. It is a technique that is, in principle, feasible for use by non-experts given appropriate packaging of the mechanisms that are involved. One such mechanism is the use of generic models to describe classes of system (see, Garlan et al., 2003) that can be instantiated to particular examples. The infusion devices described in this paper use a common generic infusion pump specification that is used by the models of the interfaces to the two device models (Harrison et al., 2013). This eases the specification problem to some degree. The properties are typically expressed in temporal logic using the variables that describe the formal system to construct propositions. The method of proof is algorithmic which means that it is more accessible to non-experts than automated theorem proving. The appropriate formulation of properties and diagnosis when a property fails is not however a straightforward process. Property formulation can be made easier by offering general property templates (see, for example, Dwyer et al., 1999) that can be instantiated to the particular requirements of the devices. Property templates and the means of instantiation are offered by the IVY tool (Campos and Harrison, 2008).

When a property fails the model checker generates sequences of states (traces) in which the property being checked fails to hold. Each trace can be seen as representing a scenario. The scenario can be further explored by human factors or domain experts. However, not all behaviours that satisfy or fail to satisfy a property are of interest. Those that are of interest are the ones that are plausible because the device or the user's training, or some other contextual factor, lead to the particular behaviour. The problem is to ensure