Modelling and systematic analysis of interactive systems

Michael D. Harrison
Queen Mary University London
and Newcastle University
UK
michael.harrison@newcastle.ac.uk

José C. Campos
University of Minho
and HASLab/INESC TEC
Portugal
jose.campos@di.uminho.pt

Paolo Masci
Queen Mary University London
UK
paolo.masci@gmail.com

Nigel Thomas
Newcastle University
UK
nigel.thomas@newcastle.ac.uk

INTRODUCTION

Two aspects of our research concern the application of formal methods in human-computer interaction. The first aspect is the modelling and analysis of interactive devices with a particular emphasis on the user device dyad. The second is the modelling and analysis of ubiquitous systems where there are many users, one might say crowds of users. The common thread of both is to articulate and prove properties of interactive systems, to explore interactive behaviour as it influences the user, with a particular emphasis on interaction failure. The goal is to develop systematic techniques that can be packaged in such a way that they can be used effectively by developers. This "white paper" will briefly describe the two approaches and their potential value as well as their limitations and development opportunities.

THE ANALYSIS OF INTERACTIVE DEVICES

Origin and Underlying Principles

This research has been concerned with the analysis of interactive devices such as medical infusion pumps, in-car air conditioning systems and flight management systems. In order to analyse these devices a tool has been developed (IVY) [3] which provides a front end to a model checker. The aim has been to develop models using a simple notation that is orientated around action, producing a textual representation of a finite state model. These device models are then subjected to systematic analysis using properties based on a set of standard templates. Counter-examples that are generated are visualised in a format that aims to ease mutual exploration with domain and human factors specialists. The IVY tool supports this approach to the analysis of interactive systems in a number of ways.

- IVY supports the editing of models. These models are described in Modal Action Logic (MAL) which is a deontic logic of actions that allows focus on the actions that the user engages in when using the device. The tool translates MAL models into SMV [6] and invokes NuSMV [5] to check properties of the model.
- IVY provides property patterns and the means of instantiating property templates associated with the patterns. The patterns are designed to probe aspects of the interactive behaviour of the device systematically in a process that is similar to the application of "usability heuristics". By using the IVY property editor it is possible to define a battery of properties to which the device can be subjected. These properties explore mode as well as the relationship between attributes of the device and what is visible about the device.
- IVY provides a trace visualizer that eases the exploration of counter-examples when properties fail. These counter-examples provide material for scenarios that can depict problematic situations in the use of the device.

This tool has also been used to explore the use of information resources to restrict analysis to paths that are "plausible" from the perspective of human factors or domain specialists [8]. Information resource constraints make explicit the information that it is assumed the user will use in order to help decide what to do next to achieve a goal.

Modelled Relationships

These models of interactive devices focus on actions. Models are designed to make explicit whether or not a state attribute or action is visible to a user. State attributes and actions have also been added to the model to capture the activities and meta-variables that reflect the use for which the device is intended. These activities and meta-variables will have been determined by studying the work that the device is designed to be embedded within. Properties checked of the model include determining the relation between attributes specified to define modes and those that indicate variables that are relevant to modes. Properties are also concerned with determining that intended goals of the device are reachable subject to resource constraints in order to generate plausible paths that can be further explored by domain and human factors experts.

Problems Addressed

The central problem of this work is to provide a systematic means of analysis of interactive systems that is objective and can be performed by analysts who are not experts in the use of formal methods. A number of standard patterns have been developed that can be instantiated for the model in question using the IVY tool. Traces of counter-examples are visualized to aid the construction of appropriate scenarios.

One example of such a standard property is whether the user of a device can recover from a wrong action. This property has a standard template and can be expressed in CTL [6] as.

$$AG(attribute = value \rightarrow AX(action1 \rightarrow EX(action2) \mathbin{\&} AX(action2 \rightarrow (attribute = value)))))$$

The Alaris infusion pump (one of the systems studied) includes chevron buttons that allow the incrementing or decrementing of data as it is entered. This property template can be instantiated by applying it to single chevron up buttons and single chevron down buttons *sup* and *sdown.* When the device is in a mode determining infusion rate entry (*entrymode=rmode)* when the infusion rate is not locked *(!rlock*) the property is expressed as.

$$AG((infusionrate = IVAL1 \mathbin{\&} entrymode=rmode \mathbin{\&} !rlock) \rightarrow AX(sup \rightarrow (EX(sdown) \mathbin{\&} AX(sdown \rightarrow infusionrate = IVAL1))))$$

IVAL1 is a meta-variable that ranges over the possible values of *infusionrate*.

Applications

Recent work has concerned the development of models of medical infusion pumps developed by three manufacturers [3,4]. These designs are the result of interesting and in some cases subtly different design decisions. Two large models have been constructed. The models reuse a common module that captures the characteristics of the underlying pump. Both models have been analysed using a battery of properties that have been instantiated for the two models so that similar properties can be checked of the two devices. The applicability of the tool in the aerospace context is also being investigated.

Limitations and Development Opportunities

MAL provides a notation that coincides well with the interaction structure of scalable systems. However the size of the models generated if interaction details are to be captured can be very large and this means that model checking is either impossible in ordinary available computer technology or requires turn around times of hours rather than minutes. Off the shelf model checkers do not exploit the multi-processor capabilities of modern computers. An alternative approach that is being explored is to translate models systematically into the specification language of PVS so that properties that are most appropriate for theorem proving, particularly concerned with the visibility of aspects of the underlying state of the model, can be proved more efficiently than would be possible using model checking. Early steps towards this work can be found in [7]. Theorem proving remains a relatively difficult procedure and therefore standard formats and procedures are also being explored, to make it easier for analysts to develop models and prove properties.

Even if MAL is well suited to model large systems, two further issues related to the modelling approach deserve attention.

- On the one hand the use of IVY requires that a model be developed for the sole purpose of carrying out the analysis. This represents a significant barrier towards adoption, especially in HCI. Solving this means either finding alternatives to the (semi-) automated development of the models, or alternative notations that better integrate into a development process. In the first case, work has been done in reverse engineering user interface code with the goal of producing models of the supported interaction [13]. However, this means that the analysis can only be performed once the system has been (at least partially) developed. An interesting alternative would be to consider the generation of models from the design artefact. Storyboards are envisaged as a possibility (indeed tools such as CogTool [11] use a similar approach). In the second case, a tabular version of the language – with lines representing actions, and columns representing attributes – could be envisaged as a more "engineer friendly" notations to express the models, see for example [1].
- On the other hand, the step from analysing the model to certifying the final system remains a challenge.

THE ANALYSIS OF MOBILE AND UBIQUITOUS SYSTEMS

Origin and Underlying Principles

This research is concerned with the analysis of systems that combine public displays and hand-held personal devices. These systems provide relevant and tailored information to users in physical environments such as hospitals, shopping malls, airports or office environments. The success of such systems depends on effective testing and user evaluation. They must be natural to users, enabling an enhanced experience of the place in which the system is situated. The evaluation of these systems is often impractical within their designed target environments. We are exploring predictive models of the interactive behaviour of these environment designs that include an understanding of the proposed context. Properties are required that relate to how the smart environment enhances or otherwise the collective user experience of complex spaces.

Modelled Relationships

Systems are described as activities in PEPA [10]. To illustrate the modelled relationships a smart system will be briefly illustrated that supports, by means of public displays, visitor routing to a particular location, where the system is aware of the location and destination of visitors.

The PEPA model consists of processes modeling the behavior of visitors, arbitrators, slot managers, slots and places. Slots and places are instantiated for each particular location. The display consists of a number of slots. The slot manager and the arbitrator, in each location, ensure that requested information is displayed and that no two slots show the same information. Groups of visitors are defined with the same starting location and final destination. For example, a visitor starting in location *a* and heading for location *d* first tries to get a place in location *a* where it is possible to see the display (*lasd* is the action of trying to acquire a place in location *a*). Once a place has been acquired the visitor engages in action *laee* to find out where to go next. The request is engaged as soon as there is an available display slot that displays the information. When the information is displayed the visitor releases the place in location *a* (action *lasu*) and receives the information (i.e. any of the matching destinations in the process *VisEdRec*). The visitor then proceeds to the indicated next location (e.g. *VisEdtoLb* means that the incomer with final destination *d* now first needs to proceed to location *b*). The arrival at the final destination is modelled by the process that remains in the state V isEdArrived forever.

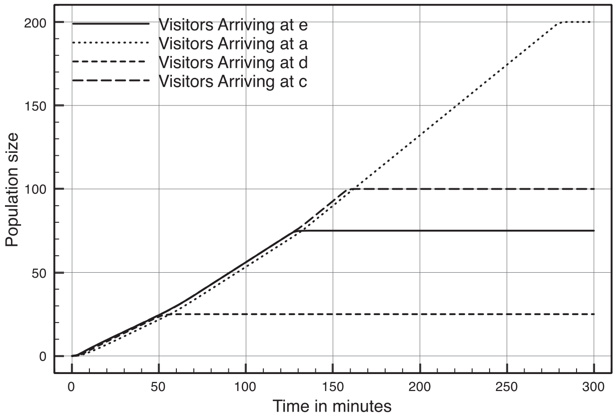
VisEytoLx = (lxsd, s).(lxey, a).(lxsu, s). VisEyRecx
VisEytoLy = (lesy, s).(lyey, a).(lysu, s). VisEyArrived
VisEyRecx = (eexle, r).VisEytoLe +
(eexle, r).VisEytoLa +
(eexle, r).VisEytoLb +
(eexle, r).VisEytoLc +
(eexle, r).VisEytoLd
VisEyArrived = (nop, a). VisEyArrived

The model of a visitor has three rate parameters, modeling the average time needed to perform the related activity. The average duration of activities is defined by their rates. Rates are assumed to be measured in minutes. So, for instance, letting *s = 10* implies that the average time a visitor needs for sitting down or standing up is 6 sec. (i.e. 0.1 min.). Rate *a = 2* models the average time a visitor needs to make a request equal to 30 seconds. The rate *r = 1* models the average time to receive the requested information equal to 1 minute. It is further assumed that visitors are arriving over a certain period of time and heading for different destinations.

A number of factors could have an impact on the person, or people, in the environment affecting their experience of it. The relevance of these factors depends on physical context. Prior to analysis they could be assessed through some form of user evaluation which can be converted into properties of the model. These factors could include:

- visibility and interpretation of display directions
- continuing visibility of directions whatever the user's location
- sense of progress towards the destination
- ability to remember the route having completed it once
- a broader sense of the building and the facilities it offers
- how long to wait before the display is relevant to them (either in terms of time or number of refreshes of the display)
- guaranteed time to arrival
- impact if many users need to recover from some scheduling change due to congestion in the environment
- a sense of congestion, that there are too many people in the surrounding space

Preliminary results have involved using fluid flow models of the systems to provide average behaviors. For example, notions of congestion can be addressed by exploring the arrival of visitors to various locations in the building.



Problems Addressed

We are interested in stochastic properties of systems involving multiple people, that is crowds of people interacting with the system [9,12]. The technique uses PEPA and a combination of fluid flow and simulation techniques.

We have also explored a mixed approach connecting formal (Petri-net based) models of the ubiquitous systems with virtual reality simulations of the target environment in order to support different levels of analysis, from empirical studies to formal verification [14].

Applications

Early applications have included the exploration of emergency egress in an office building and out-patient behaviour in the context of a hospital department.

Limitation and Development Opportunities

This work is in early stages. The results relating to emergency egress are promising [12] and have produced results that are consistent with other simulations of the same emergency egress problem. There are several opportunities for future work and limitations to overcome. For example,

- The work is based on a stochastic process algebra (PEPA) [10] and the notation is not conducive to the expression of large models. Relatively small models are quite unwieldy to represent. We are however working on alternative approaches to developing simulations that may be more expressive and scalable.
- The semantics of the approach is based on exponential memory-less distributions and this approach may not be the most appropriate for the kind of problems we wish to tackle. PEPA uses fixed rates rather than functional rates and many of the scheduling problems, that can be solved dynamically using a smart environment to improve flow, require functional rates
- We need to quantify the characteristics of these environments that capture the experience that people within the environment. We can measure flow in terms of delay or slack time for example, but how do we quantify the frustration that users suffer? There are models of emotion that might help us here [15].

ACKNOWLEDGMENTS

José Campos funded by the ERDF – European Regional Development Fund – through the COMPETE Programme, and by National Funds through the FCT – Portuguese Foundation for Science and Technology – the APEX project PTDC/EIA-EIA/116069/2009. Michael Harrison and Paolo Masci are funded by the CHI+MED project: UK EPSRC Grant Number EP/G059063/1. Nigel Thomas is funded by the AMPS project: Analysis of Massively Parallel Stochastic Systems EPSRC EP/G011389/1.

REFERENCES

1. Abowd, G.D., Wang, H-M. and Monk, A.F. A formal technique for automated dialogue development. DIS'95: Proceedings of the 1st Conference on Designing Interactive Systems: processes, practices, methods and techniques. ACM Press. pp. 219-226. 1995.
2. Campos, J.C. and Harrison, M.D. Systematic analysis of control panel interfaces using formal tools. In N. Graham and P. Palanque, editors, Interactive systems: Design, Specification and Verification, DSVIS'08, volume 5136 of Springer Lecture Notes in Computer Science, pages 72–85. Springer-Verlag, 2008.
3. Campos, J.C. and Harrison, M.D. Interaction engineering using the IVY tool. In G. Calvary, T.C.N. Graham, and P. Gray, eds, Proc. ACM Symp on Engineering Interactive Computing Systems, pp 35-44. ACM Press, 2009.
4. Campos, J.C. and Harrison, M.D. Modelling and analysing the interactive behaviour of an infusion pump. Electronic Communications of the EASST, 5, 2011.
5. Cimatti, A., Clarke, E., Giunchiglia, E., Giunchiglia, F. Pistore, M., Roveri, M., Sebastiani, R. and Tacchella, A. NuSMV 2: An Open Source Tool for Symbolic Model Checking. In K. G. Larsen and E. Brinksma, editors, Computer-Aided Verification (CAV '02), volume 2404 of Lecture Notes in Computer Science. Springer-Verlag, 2002.
6. Clarke, E.M., Grumberg, O and Peled, D.A. Model Checking. MIT Press, 1999.
7. Doherty, G.J., Campos, J.C. and Harrison, M.D. Representational Reasoning and Verification. Formal Aspects of Computing, 12(4):260-277. 2000.
8. Doherty, G., Campos, J.C. and Harrison, M.D. Resources for situated actions. In N. Graham and P. Palanque, editors, Interactive systems: Design, Specification and Verification, DSVIS'08, volume 5136 of Springer Lecture Notes in Computer Science, pages 194–207. Springer- Verlag, 2008.
9. Harrison M.D. and Massink M. Modelling interactive experience, function and performance in ubiquitous systems. In: Electronic Notes in Theoretical Computer Science: 4th International Workshop on the Practical Application of Stochastic Modelling (PASM). Imperial College, London: Elsevier. 2010.
10. Hillston, J.A. A compositional approach to performance modelling. Cambridge University Press, 1996.
11. John, B., Prevas, K., Salvucci, D., & Koedinger, K.. Predictive Human Performance Modeling Made Easy. Proceedings of CHI 2004. ACM, 2004.
12. Massink M, Latella D, Bracciali A, Harrison MD and Hillston J. Scalable context-dependent analysis of emergency egress models. Formal Aspects of Computing: Volume 24, Issue 2, Page 267-302. 2012
13. Silva, J.C., Silva, C., Goncalo, R., Saraiva, J. and Campos, J.C. The GUISurfer tool: towards a language independent approach to reverse engineering GUI code. In Proceedings of the 2nd ACM SIGCHI Symposium on Engineering interactive computing systems, pages 181-186. ACM. 2010.
14. Silva, J.L., Campos, J.C. and Harrison, M.D. Formal analysis of Ubiquitous Computing environments through the APEX framework. In ACM Symposium on Engineering Interactive Computing Systems (EICS2012). ACM. 2012. (accepted)
15. Steunebrink, B.R., Dastani, M. and Meyer, J-J Ch. A formal model of emotions: integrating qualitative and quantitative aspects. In M. Ghallab et al. (eds) ECAI 2008 IOS Press Pages 256-260. 2008.