

A Category Theory Approach to HCI

David England
Liverpool John Moores University
Byrom St, Liverpool, L3 3AF, UK
d.England@ljmu.ac.uk

In this paper we revisit Category Theory as a modelling tool for Human Computer Interaction. We first review past approaches and limitations to formal methods in HCI. We then introduce a category theory with a tutorial example, before moving onto a more complex example. We end by discussing the further implications for theories of HCI.

Category theory, formal methods,

1. INTRODUCTION

Formal modelling of Interaction has had a long history in HCI [Dix 1991]. However, in the last 20 years it has been side-lined in mainstream HCI as empirical usability approaches predominate. In some senses this is understandable. HCI is a young, interdisciplinary practice, with little in the way of firm theory with which to model. Some of the earlier cognitive modelling approaches have fallen out of favour as interaction becomes more social. Similarly, more mathematical approaches have been restricted to niche research, such as, safety-critical systems. In addition some of the mathematical modelling approaches have been borrowed from other areas of Computing Science. As such they often require additional effort, both in learning and in deployment, in order to model interaction scenarios. The focus on user centred design and research has also pushed mathematical approaches into the background. Any technique which is seen as not providing evaluative feedback during an iterative, development cycle is seen as flawed.

Our view is that this side-lining of the Mathematics of HCI is short-sighted. At heart we believe HCI is an engineering discipline with the aim of improving the quality of human interaction with and through technology. The current focus on user centeredness and social interaction, though superficially providing answers to some challenges, is doomed to fail. This failure will arise as systems become every more complex. We already have systems that are composed of multiple, distributed components involving hundreds of millions of users. It is impossible to empirically test the value of such systems. Indeed many systems available to us seem to be in constant beta (and even alpha) release. And there are howls of anguish each time

one of these beta level systems undergoes a live change, unsettling users and losing customers. Beyond the interaction problems, such permanent-beta systems also exhibit constant problems with robustness and security. As many of us rely on such systems for our daily lives, these disruptions can be more than minor irritations.

How might a more formal approach help with these challenges? The answer is the same as that given for the support of formal methods in general: by capturing knowledge (of HCI) in a more rigorous framework that can be re-applied to future problems. The counter argument is that formal models do not capture context. Thus the knowledge apparently gathered cannot be reapplied in new scenarios without substantial effort. Suchman [Suchman 2006] makes much of this argument in "Plans and Situated Actions" where the ideas of algorithm and plans are seen as secondary to the situation of study. However, here Suchman is making an argument, primarily, for the social study of technology rather than the research and design of technology itself. What if there was a mathematical approach that captured context and situation as easily as plans?

Past approaches to the use of formal methods in HCI have used process-based algebras like CSP [Alexander 1987] and Petri Nets [Silva 2012]. This highlights one of the problems of using adopted formal methods. In order to use the method we have to see our HCI problem through the lens of the method we have adopted. So process methods see interaction as a process or set of process; object oriented methods put classes and objects first, and so forth. What we require is a method, or methods, that allow us to model our HCI problems in an appropriate manner and at the correct level of abstraction.

2. CATEGORY THEORY

Our argument in this paper is that Category theory is a prime candidate for the formal modelling of interaction. At its simplest Category theory is the mapping of one group of mathematical objects to another group. It is a general mathematical theory of structures and of systems of structures [Lawvere 2009]. As such it is said to supersede Set Theory as a general approach to mathematical modeling. Category theory models the mapping of one system to another. The power behind the mapping is the general set of rules that can be used to prove the correctness of the mapping *regardless* of the type of the Category under consideration.

We have already since the value of Category Theory in Computing in the guise of Functional Programming [Bird 1998]. Here category theory is used to establish the correctness of type systems to ensure that programmes behave as expected. However, Functional Programming treats input and output (i.e. all Human Interaction!) as a side effect in the formal system. As such, HCI is said to be outside the formal system being modeled. Given the massive growth in interactive and social computing, leaving the majority of a system *outside* the formal model is a major restriction of the functional approach.

The answer to this problem is to go back to the mapping power of Category Theory. If we consider HCI to be primarily a mapping problem (between and through the artificial system to users), we can begin put these “side effects” to the fore.

3. A TUTORIAL EXAMPLE

For our first example let us consider the apparently simple case of a screen layout. We are going to assume that the quality of the screen layout is a function of Fitts’ Law and Signal Detection theory. Optimising for Fitts’ law means the interactive elements are grouped to minimise input device movement and error, whereas Signal Detection theory tells us to optimise the signal elements (foreground) of the screen over noise elements (background and decoration). For simplest of this example let us assume all our elements are rectangle. Our screen design therefore consists of rectangle components whose position conform to Fitts’ law and Signal detection theory. Therefore our hypothesis would be that there exists one or more sets of screen elements positions that optimise user interaction. So we have created a filter on *all* possible designs to rule out those which are non-optimal.

In Category theory terms we have to express categories as nodes in a graph which are connected by arrows which represent mappings or functions.

Our first mapping, f , is from non-optimised Fitts’ law positions, NOPFL to optimised positions OPFL

$$(f) \text{ NOPFL} \rightarrow \text{OPFL}$$

Our second mapping, g , is from non-optimised Signal Detection theory positions, NOPSD, to optimised signal detection theory positions, OPSD

$$(g) \text{ NOPSD} \rightarrow \text{OPSD}$$

In Category theory the first object is the domain, and the second element the co-domain of the mapping. If the types of domains agree, we can combine functions applied to them. So in our above example

$$(f \bullet g)$$

means the composition of our two functions to provide a final co-domain which is the set of element positions optimised for both sets of criteria.

Diagrammatically we would show this as

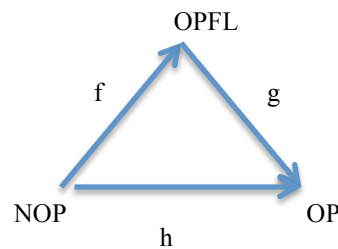


FIGURE 1

Where $h = (f \bullet g)$ and OP is the resulting co-domain of all optimised positions.

At this level of discussion we can consider positions in the abstract. They could be pairs of coordinates for rectangles in a simple display, centres of more abstract shapes, or positions of 3D objects in a virtual reality display. Having established the principle of an optimised display we are free to examine concrete examples so long as the starting domains for our optimisation functions are the same. We can also introduce further requirements or criteria as further mappings which are composed with our existing functions to produce a further level of optimisation, or requirements matching.

Here we can see the beginnings of a general Category approach to HCI. We first identify our target co-domains and the requirements (mappings

or morphisms) that are necessary to produce them. We can apply some of the general rules of category theory such as associativity and identity functions to prove our morphisms are valid, i.e. the structures of our domains are preserved. Now we can do this on a project-by-project basis, applying new requirements as we discover them, or we can apply our category validations on more generic entities like heuristics or patterns, to prove that they generate valid categories in a given context.

4. UBIQUITOUS EXAMPLES

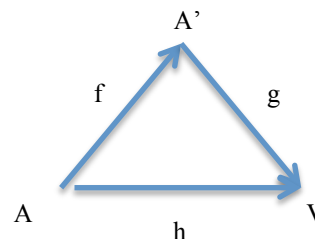
Let us consider a more complex example using social and location-based media. Milner [Milner 2009] has provided one basis for modelling ubiquitous computing scenarios using Bigraphs. In Bigraphs the problem is described by a link graph of the information content of the problem and an associated space or place graph with captures the physically based rules of the same problem domain. The space graph defines the scope and physical rules of action in the information “space”. Thus we can define a range of scenarios of mixed realities, true virtual realities and augmented realities in a rigorous way. Bigraphs could also been seen as a formalism of the Spatial Model [Benford 1993]. Bigraphs are of course mainly based on Category Theory, but by stepping back a little we can ask; do we need separate link and space graphs? Can we use the category of algebraic topology as ours spaces, and the rest of the information-related categories (graphs, sets, reactive systems etc.) as our “link” graphs? This would seem to be a broader approach that would cover many scenarios in social and location-based interaction.

As an example let us consider formalising some aspects of the Spatial Model but with an eye to current challenges of social media and of location-based interaction. The early spatial model was aimed at supporting human communication protocols in shared virtual environments. Each person’s avatar was surrounded by a volume or aura. If the avatar’s aura intersected with the aura of another object that would determine how they could mutual interact. Avatars also had a focus mechanism which controlled how they perceived their environment through various communications media. Objects in the world could act as adapters. Avatars coming into the zone of an adapter would have their communications scope altered. Thus a podium adapter would give a person greater attention in a group interaction. A table adapter would limit communications to that group of avatars around the table, and so on. In this version the spatial model is mostly concerned with the

topological arrangement of avatars and objects in managing intercommunication and perception. It did not, in a strict sense, have the equivalent of a link graph. So again, taking a step back, using category theory, we can start to re-examine the Spatial Model, with a view to making it a general model of interaction, applicable to shared spaces, whether they are physical or virtual.

Suppose we have a set of medical staff who perform treatment in the community. They carry with them their treatment plans for the day. Their patients maybe in their own homes, in a clinic or temporarily in hospital. As part of the treatment requirements there are a number of safety protocols in place, which determine when, where, with whom and to who treatment can be carried out. The staff are guided by a treatment tablet outlining their schedule for a period of time. So we can start to see some of the requirements for the system, which are determined by a set of individual treatment situations.

We have a domain of available treatments, A , and a co-domain of valid treatments, V , in a certain situations. The morphisms from available to valid will include rules about location, f , and co-location of staff, g .



So as a strongly controlled example, we might determine that the administration of morphine, when done in a patient’s home (f) has to be done in the presence of two people, one of whom is a doctor (g). So we have a location (spatial) determinant and a linked-based determinant to this scenario. In Spatial model terms, the link (the second person) is an adapter to the behaviour of the first person.

We might have less restrictive administration protocols for other drugs and/or locations. So, morphine can be administered by a doctor or nurse in a hospital setting. Penicillin can be administered by a doctor alone in a home or clinic setting, or a nurse alone in a hospital setting.

So having established the *abstract* template of possible actions to permitted actions, together with their morphisms, we can determine the parameters of what should happen in *particular* interaction scenarios.

In a less critical scenario we might consider the relationship between privacy and visibility in social media. This is often a source of confusion amongst social media users, who may not be aware of the information they are sharing with others. Facebook is often cited as an example where public visibility is the default, and where the user's comprehension of their visibility is not clear. Google+ is slightly more transparent with its circles metaphor. There are similar problems with online learning environments such as Blackboard. Here the problem is one of mutual visibility. What can be seen from the tutor's viewpoint may not be seen from the student's, and vice versa. Visibility and reachability are often cited in standard HCI texts as desirable properties of an interface, but the question is: in what context?

Again we can step back and consider what are our domains, co-domains and their morphisms. We have a domain of available information provided by people to their interactive systems. We have co-domains which are selections on the domain which we may or may not want to make visible to certain other classes of user(s). The interaction design challenge is what morphisms we need to provide and make visible to users can they can have full and coherent control of their information. The range of morphisms would be from fully visible to all to visible to myself. And the further design challenge would be to allow users to preview the implications of their visibility choices before they went live. No current social media system, that I am aware of, does this. Visibility and Privacy is often a case of trial and embarrassing error.

4. DISCUSSION

How then can we move forward with Category theory in HCI research and design? We could take a standard project approach, using Category Theory as the framework for our methodology, be it phased, spiral or iterative. The role of Category Theory would be in capturing the abstract and concrete categories of information discovered in requirements gathering. We would also determine the morphisms in moving from general categories to the particular as we refine our requirements, designs and implementations. Most development approaches to HCI involve dealing, unsatisfactorily, with different levels of abstraction simultaneously. We move from frequently general principles to particular details and back again. Very often in this process, those requirements which are a priority for

users get lost. So there is role for category theory as a documenting and checking mechanism across the stages of development.

Having used category theory to check our requirements and designs we can also use it to connect our designs to evaluation techniques. Currently there is an intellectual gap between design and evaluation. Very often the results of evaluation are messy and difficult to reflect back to re-design proposals. Results of evaluation are often presented as a discursive account and interpretation of some set of statistics. However, if we were to frame our evaluation questions as validating our categorical choices, we would have a firmer basis for providing re-design proposals.

The same concepts could be applied to the project of RepICHI – supporting the replication of results in HCI experimentation. This project currently faces many challenges, some of which are socio-technical, but the most clear technical challenge is the lack of any clear theoretical basis on which to make any comparisons between past and current experiments. How can we make the conceptual links between different experiments when their contexts, subjects groups and even technologies of study are different? Here again category theory has a role to play. If we can find similar information domains and morphisms between domains in different experiments, we can begin to have a basis for comparison.

Adaptive user interfaces have long been an unfulfilled goal of HCI. The main challenge has been systems that adapt either trivially or intrusively. Autonomic computing has been proposed [England 2009] as a means of systems (and interfaces) adapting themselves to changing contexts. The issue here has been the modelling meta-language used to express the degree and dimensions along which adaption can take place. Again we would propose a category theory approach to providing that meta-language, using abstract categories as guides for adaption.

In Category theory there are families of categories of different meta-domains such as sets, topology, fields, groups and so on. It requires further investigation to see how valuable these are in the modelling of human computer interaction. In the long run what we probably need to develop is a Category of Interactions that applies specifically to the dynamics of Human Computer Interaction. This will be a long-term project in formulating a formal framework for HCI. This would also lead to the development of Category based development tools which allow requirements specification, validation and interface production.

4. CONCLUSIONS

As it stands HCI lacks a unifying framework in order to bring its different knowledge sources together. Indeed HCI could be said to be fragmenting as different areas emerge and present what seem as new and unique challenges. This fragmentation is probably damaging to HCI in the long-term. Category theory and its evolution offers such a unifying framework, bringing different domains of knowledge together in a rigorous way.

5. REFERENCES

Alexander H, (1987) *Formally-based Tools & Techniques for Human-computer Dialogues*, Ellis Horwood.
Benford S (1993), Fahlen L, *A Spatial Model of Interaction in Large Virtual Environments*, Proceedings of ECSCW '93, Springer
Bird R (1998), Wadler P, *Introduction to Functional Programming*, Prentice Hall.

Dix A (1991), *Formal Methods for Interactive Systems*, Academic Press.
England D (2009), Randles M and A. Taleb-Bendiab A, "Runtime User Interface Design and Adaptation", *In Proceedings of Human Computer Interaction 2009 (HCI2009)*, Cambridge, UK, 1st-5th September, 2009
Lawvere F W (2009), Schanuel S H, *Conceptual Mathematics, a first introduction to categories*, 2nd ed., Cambridge University Press.
Milner R (2009), *The Space and Motion of Communicating Agents*, Cambridge University Press.
Silva J L (2012), Campos J, Harrison M, *Formal Analysis of Ubiquitous Computing Environments through the APEX frameworks*, proceedings of EICS 2013, ACM, New York.
Suchman L, (2006) *Human-Machine Reconfigurations: Plans and Situated Actions*, 2nd ed, Cambridge University Press.