Distorting Design: Unevenness as a Cognitive Dimension of Design Tools

Martin Stacey

Computing Department
The Open University
Milton Keynes MK7 6AA, UK.
M.K.Stacey@open.ac.uk

Abstract

All design tools are *uneven*: they make some designs harder to produce than others. They bias designers towards producing designs that the tool makes easy. This paper gives an informal definition of *unevenness*, and argues that it is a *cognitive dimension* of design tools. Unevenness is an emergent property of other cognitive dimensions, primarily *closeness of mapping* and *viscosity*. Important sources of unevenness are: imposed structure (forcing the user to fit the system's restrictions); distortion (changing the design to fit the system's restrictions); default rigidity (the difficulty of changing parameters); accessibility (the difficulty of finding out how to use the system); prepared paths (having designs and templates provided); and tediousness.

Keywords: HCI, design, CAD, cognitive dimensions.

Published in G. Allen, J. Wilkinson and P. Wright (Editors)
Adjunct Proceedings of HCI'95: People and Computers.
The 10th Annual Conference of the British HCI Group.
University of Huddersfield, Huddersfield, UK, 29 August - 1 September 1995.

1 Introduction

Every design tool makes doing some things easy and other things difficult, whether the tool is a CAD system, text formatter, drawing package or pencil. Very often creating one result is easy (like drawing a line 2 points thick using MacDraw), when an equally simple and obvious result is impossible (like drawing a line 3 points thick). A design tool is *uneven* when it makes achieving one result harder than another, when both are equally simple and obvious. Designers can only produce designs that their tools make possible, and they are channelled into producing designs that their tools make easy, so the unevenness of design tools has a powerful effect on the results of the design process.

2 What is Unevenness ?

The *Unevenness* of a system is the extent to which it makes some designs easier to produce than others. As the difficulty of producing a design depends on the intrinsic size and complexity of the design, unevenness can only be defined in terms of the relative difficulty of producing designs that are equally simple and obvious. Simplicity and obviousness are subjective: they depend on different designers' conceptions of the domain - the complexity of forms, and how designs are make up of elements and relationships between them. So unevenness is the result of the mismatch between the simplicity and obviousness of designs (in terms of users' mental combinations of conceptual elements) and the simplicity and obviousness of the combinations of operations needed to produce them.

The operations and design representations provided by a design tool determine how easy it is to reach different parts of the space of imaginable designs. Unevenness produces slopes in the space of possible designs, when producing some designs is harder than producing others that are intrinsically equally simple; or cliffs, when producing some imaginable designs is made completely impossible by the tool.

All design tools are uneven. Designers' conceptions of elementary units and relations, and operations for changing and combining them, include more levels of abstraction and more subtle relationships than the objects and operations provided by any feasible tool. Moreover, the costs of using the primitive operations provided by a tool to produce design elements may not be closely related to the conceptual simplicity of these design elements. For example, a pencil makes drawing freehand lines very easy, straight lines harder, and doing duplicate-region or mirror-image-region operations extremely costly.

Simple tools are very uneven: they provide only one or a few primitive operations, that allow the user to do a few things very simply, and other things with great effort or not at all. This either restricts the users' choice of conceptual operations and objects, and so limits the range of designs possible (for example, a simple text formatter), or it requires the use of large numbers and complex combinations of primitive operations to create conceptually simple forms (for example, a pencil). So there is a trade-off between simplicity and restrictiveness (producing cliffs in the design space) and a trade-off between simplicity and effort (producing slopes in the design space). Adding features and parameters to a tool has two kinds of costs: complex tools are harder to learn, and each action is harder or slower to make because it has to be selected from a wider range of possible alternatives. Compare a pencil with a three dimensional surface modelling package.

3 Unevenness as a Cognitive Dimension of Design Tools

Thomas Green (1989, 1991) has argued that computer systems and similar artefacts differ along a number of *cognitive dimensions*, which influence how their users think about the systems and about their tasks, and so influence their usability. This paper extends Green's cognitive dimensions framework by introducing the cognitive dimension of unevenness. Unevenness is an emergent property of other cognitive dimensions, primarily the *closeness of mapping* between the structures and operations provided by the tool and the concepts and actions of the domain. The unevenness of a system is also influenced by its *viscosity* - the effort involved in making changes to a design. Other cognitive dimensions have more subtle effects on design. *Imposed guess ahead* - the consequence of forcing the user to do things in a particular order - makes producing designs that fit the system's structure easier than producing designs that don't. Systems that create designs with *hidden dependencies* bias designers towards retaining unintended effects, and away from using structures and operations known to cause problems.

4 Sources of Unevenness

A number of aspects of the design tool and its accompanying documentation contribute to the nature and degree of its unevenness.

Imposed Structure. Many target domain parameters or desired effects are variable to a fine degree of resolution, but design systems only provide a few discrete variants to choose from. For example, most drawing packages only provide a few possible thicknesses of line, and others are impossible or extremely difficult to obtain; the packages don't provide a way to get the variation inherent in the domain. The second form of imposed structure effect is produced when the target domain has an inherent structure which doesn't match the structure imposed by the system. For example, some CAD systems for pattern making in garment design provide curve drawing procedures that use natural splines; these restrict the user to a limited range of curves, which provide an unsatisfactory match to the limited range of curves the designers really want.

Distortion. The user's design is distorted if the system modifies it to fit its own internal constraints. Distortion can come from a conflict between the granularity of the domain and the that of the system, for example when a smooth line is digitised to a set of point coordinates or pixels that are not smooth when looked at closely. Similarly distortion can come from the system forcing the user into using one of a small number of possibilities when the target domain is more varied, for example when a hand-drawn line is turned into a set of points joined by straight lines or splines.

Distortion is closely related to imposed structure but not the same thing: altering your choice rather than telling you what to choose. This distinction is about whether the system constrains the users' actions, or the meanings those actions can have. In addition, manufacturing processes and the properties of materials produce deviations between the CAD system's representation of a design and the actual physical object. The location of these different *closeness of mapping* effects in the design process is illustrated by the figure.

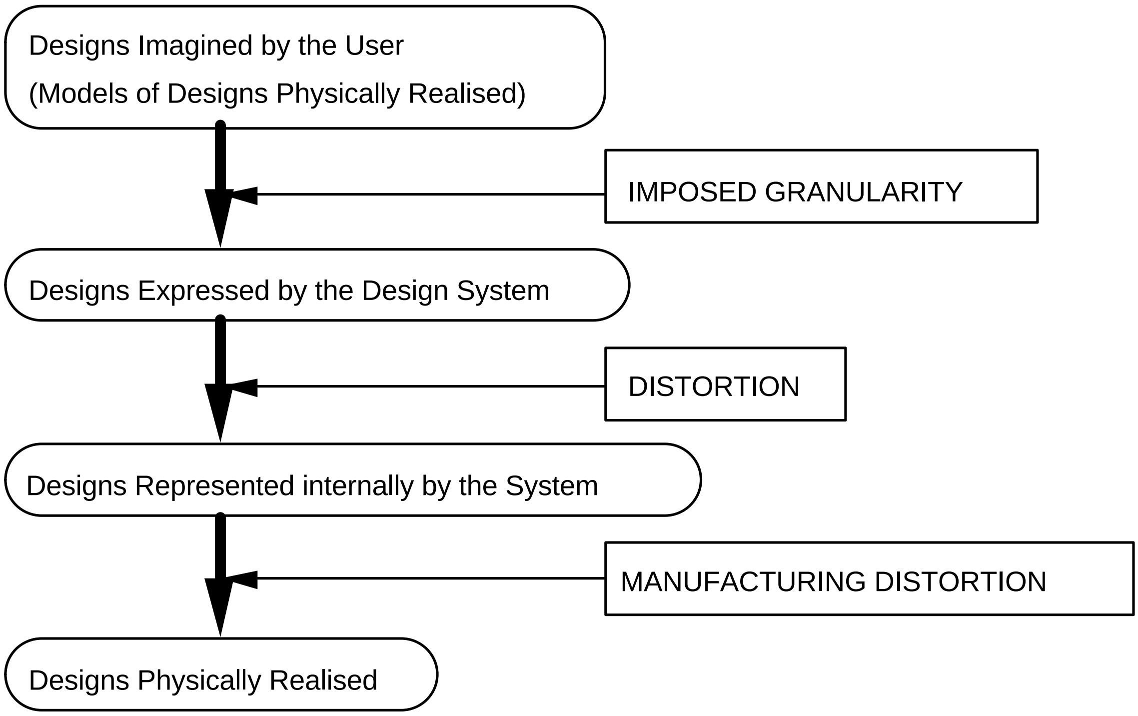

Figure: Distortion effects produced by design tools

Simulation. Simulation in this sense is the use of multiple actions or design elements to get something that is a primitive action or element in another representation. Unevenness results from simulation if the user of a design tool needs to use more complex groups of primitive design elements or operations to do or create something that is a simple entity in the target domain, and the tool provides easier ways to do or create other things that are comparably obvious or complex. This is a *closeness of mapping* effect. The more complex operations have more chance of error, are harder to remember and are less individually discriminable (Reason, 1990).

Default Rigidity. This is the amount of effort involved in changing the system's default behaviour, either globally for an entire job or locally for one operation. Many defaults cannot be altered at all. When changing the default behaviour of the system is hard work and might produce unexpected side effects, the user is strongly discouraged from trying it. This is a *viscosity* effect. A steep slope becomes a cliff for inexpert users.

Accessibility. You need to *know* how to do things. Unevenness is imposed by differences in the users' knowledge of the available operations, independently of how hard they are intrinsically. This is influenced by the users' training and the adequacy of the manuals or help facilities. Unevenness resulting from accessibility problems is a feature of complex programming languages like C++.

Prepared Paths. It is easier to produce designs for which part of the work has already been done. Unevenness results from the availability of templates or examples for creating designs that are difficult or time-consuming to produce, especially using tools that make juxtaposing available parts of designs easier than modifying them.

Tediousness. Primitive operations that are easier than others will be preferred when the latter are not required, and designers are deterred from making difficult changes. Examples of tediousness factors are the amount of typing, the numbers of parameters a command requires, and the size of a screen button needed to select it. This is an element of the *viscosity* of the system.

5 Distorting Design: The Influence of Unevenness

The unevenness of design tools has a powerful influence on the designs they are used to produce. Clearly, designers are unable to create designs that are impossible to generate with the tools they are using, even if they can imagine them perfectly well. Slopes in the design space have a more subtle effect: designers tend to choose to do what is easy to do with the system they are using, rather than push the system's capacities. This has been observed by Devane (1992) in textile design, and by us in knitwear design (Eckert & Stacey, 1994; Eckert, in preparation). Thus the system pushes the users towards producing standard designs, that only vary across a small part of the space of possible designs.

The influence of unevenness depends on the demands of the task and on the personality of the user. When designers have a choice in how much of the system's functionality they use, unevenness of difficulty has a much greater limiting effect. This effect depends on the users' personalities and their perceptions of unevenness: how willing they are to try to find out how to get difficult-to-achieve effects out of a design tool. System developers and working designers often have very different views of what is difficult to do. Many of the latter do not enjoy playing with systems, and only try to find out what the system can do when they have an urgent reason for achieving a result. Some users are actively afraid of exploring systems and making mistakes. In consequence non-standard actions incur learning costs and are much harder for users than system developers may expect, so unevenness effects are magnified, especially those due to accessibility.

Some CAD/CAM systems provide powerful methods for producing a limited range of designs, plus the ability to do lower-level editing or programming to achieve different or more complex effects. The consequence of this, as we have observed in the case of knitting machine programming (Eckert & Stacey, 1994; Eckert, in preparation), is that the system's users are deskilled: they never learn to use the lower level programming facilities for the easy cases, so they lose the ability to create the more unusual or complex designs that the CAD systems permit in principle. A very steep slope becomes a cliff. This amplifies prepared paths effects: Eckert (in preparation) observes that the example programs provided by the knitting machine manufacturers have a significant influence on knitwear design, where some very fancy features can now be manufactured but the only examples actually produced are copied from the machine manufacturers' examples.

The pressure of uneven systems towards producing standard designs can accelerate designer burnout: In some important design activities, for example in the clothing industry, novel and unusual designs are needed, but producing them gets harder and harder as designers become expert, that is, they develop effective standard procedures, and stop needing to explore and make mistakes on the way to generating solutions to their problems. Making the non-standard moves required to produce innovative designs becomes harder as the obvious moves become more obvious. In the knitwear industry designers expect to get stale after a couple of years designing for the same market, and to change jobs to get a different set of requirements (Eckert & Stacey, 1994).

Acknowledgements

This paper owes much to discussions with Thomas Green and Claudia Eckert. They and Helen Sharp made valuable comments on earlier drafts. The work has been supported by EPSRC grant GR/J48689 to George Rzevski, Helen Sharp and Marian Petre at the Open University, for the FACADE Project.

References

Devane, J. (1992). Training and Education in CAD clothing/textile systems.
In W. Aldrich (Ed.) *CAD in clothing and textiles, a collection of expert views.*
Oxford, UK: BSP Professional.

Eckert, C.M. (in preparation). *Intelligent Support for Knitwear Design*. PhD Thesis, Design Discipline, The Open University, Milton Keynes, UK.

Eckert, C.M. and Stacey, M.K. (1994). CAD and the Division of Labour in Knitwear Design. In A. Adam, J. Emms, E. Green and J. Owen (Eds.) *Women, Work and Computerization: Breaking Old Boundaries - Building New Forms.*
Amsterdam, The Netherlands: North-Holland.

Green, T.R.G. (1989). Cognitive dimensions of notations. In A. Sutcliffe & L. Macaulay (Eds.) *People and Computers V*. Cambridge, UK: Cambridge University Press.

Green, T.R.G. (1991). Describing information artefacts with cognitive dimensions and structure maps. In D. Diaper and N.V. Hammond (Eds.) *Proceedings of "HCI'91: Usability Now", Annual Conference of the BCS Human-Computer Interaction Group.* Cambridge, UK: Cambridge University Press.

Reason, J. (1990). *Human Error*. Cambridge, UK: Cambridge University Press.