BEHAVIOUR & INFORMATION TECHNOLOGY, 1995, VOL. 14, NO. 4, 251–263

The Cockpit Metaphor

LYNNE COLGAN

Admiral, 200 Woodlands Court, Ashridge Road, Bristol BS12 4LB

ROBERT SPENCE

Department of Electrical Engineering, Imperial College, Exhibition Road, London SW7 2BT

PAUL RANKIN

Philips Research Laboratories, Cross Oak Lane, Redhill, Surrey RH1 5HA

Abstract. Engineering design is increasingly being supported by automatic procedures capable of improving a design but which, nevertheless, require human guidance if they are to be successful. Such guidance requires an effective interface. One such interface, recently implemented within a complex engineering design tool, is based upon the Cockpit Metaphor which is the subject of this paper. The metaphor was invented by domain experts and a psychologist, not in response to a commission but as an innovative statement of a fruitful path which future engineering design tools might follow. This paper describes the context of the Cockpit Metaphor, the requirements influencing its incorporation in the Cockpit interface, the evaluations carried out, and the research issues raised.

1. Engineering design

1.1. *Parameters, properties, and objectives*

Engineering design—the design of bridges, automobile engines, and electronic products, for example—has much in common with everyday activities such as the design of a dinner party or the layout of a newly furnished room. There is always a set of things, called **parameters** (figure 1) that you, the designer, are free to choose: for a dinner party these would include the guests, the seating plan, the actual courses and the wines (to assist those readers unfamiliar with engineering design, important terms are presented in **boldface**). The reason for making such choices with care is that these parameters influence the **properties** or **performance** of whatever is being designed, and it is the performance with which you are concerned. Usually you, the designer, will have a single overall **objective** in mind (for example, 'an intellectually stimulating dinner party'), an objective whose achievement depends, in turn, on the achievement of **sub-objectives** of varying importance or **weight**: the quality of the conversation and the taste of the food in the case of the dinner party. The relationships shown in figure 1 and described in the context of dinner party design are identical to those for electronic or structural engineering design, for which appropriate annotation is presented in figure 1.

1.2. *Difficulties*

Unfortunately, there is no way of proceeding *directly* from a specification of desirable performance to the parameter values that will satisfy that specification precisely or adequately. Instead, the designer must gradually move towards an acceptable design by an iterative process involving successive stages of proposal and evaluation.

Design is not easy, for many reasons. One difficulty arises from the typically large number of parameters whose values must be chosen: there may potentially be hundreds or thousands or even more, and even when judiciously selected on the basis of experience there may still be as many as ten. Similarly, the number of performances/properties of interest may run into the hundreds.

A second difficulty arises from the normally very complex nature of the influence of the parameters on the properties: each parameter will typically influence many properties. A third difficulty, directly related to the second, is that trade-off (conflict) situations are often present in which one cannot simultaneously improve two properties. This in turn makes

0144-929X/95 $10·00 © 1995 Taylor & Francis Ltd.

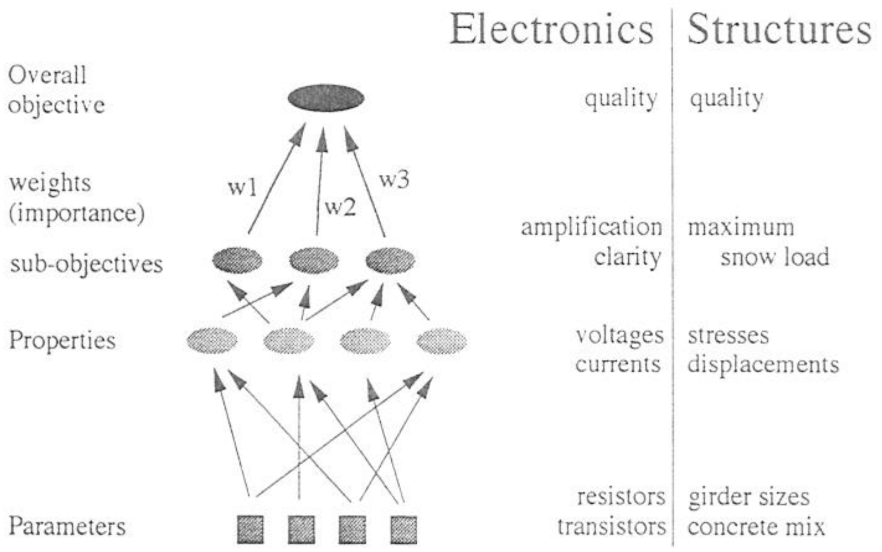


Figure 1. The hierarchical nature of engineering design.

it difficult to choose weights representing the importance of various properties. A fourth difficulty is presented if there is no easy and rapid way of predicting the properties that will be exhibited for a particular choice of parameter values.

Collectively, the difficulties associated with the design process are considerable. Sometimes they are overwhelming, as evidenced by the disintegration of the Tacoma Narrows Bridge, the failure of the Piper Alpha oil rig, and the collapse of the Ferrybridge cooling towers. Tools are therefore needed to minimize or overcome these difficulties, and are the context of this paper.

1.3. *Simulation*

Although the general process of engineering design, involving successive iterations of proposal and evaluation, has remained unchanged for many decades, certain *steps* in that process have been transformed almost beyond recognition. The development of mathematical models and, most dramatically, the availability of powerful computers that can efficiently execute these models and thereby predict the effect of parameters on properties (figure 2), has allowed the laborious construction and measurement of an artifact to be replaced with conveniently arranged **simulation**. Examples of the extremely sophisticated simulators now available include HSPICE and SABER for electronics and NASTRAN for structures. Henceforth in this paper we assume, without further discussion, the availability of a simulator for engineering design in general and, in particular, for the process of electronic circuit design which is the focus of this paper. Thus, the example of dinner party design, used as a familiar context in which to introduce terminology, would be excluded from consideration!

1.4. *Manual iterative design*

The combination (figure 3) of a powerful simulator and a (normally graphics-based) interface allowing its fluent application is now a commonplace tool in engineering design offices. Such a tool supports the iterative 'propose and evaluate' design process which evolved long before simulation became available. Because this paper demonstrates a potentially radical complement to this process, it is useful first to summarize, briefly, the essential characteristics of the conventional manual approach to engineering design.

Following the receipt of a **performance specification** from a customer, the artifact designer would first, on the basis of past design experience, suggest a possible design: in other words, an artifact would be described that was thought to

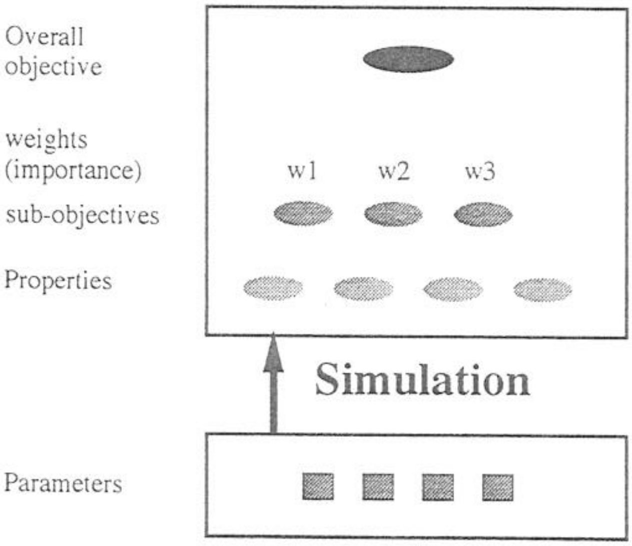


Figure 2. Simulators predict performance from a knowledge of parameter values.

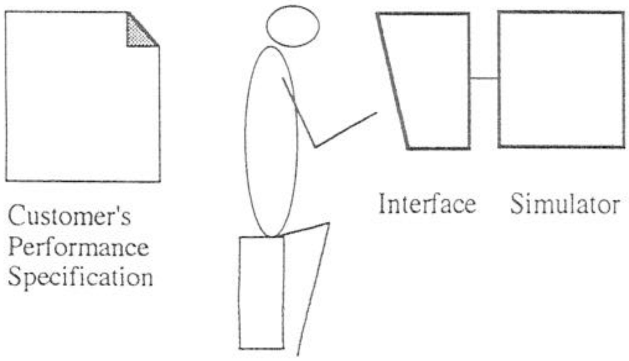


Figure 3. The use of a simulator to manually design an artifact in response to a customer's specification.

have a reasonable chance of exhibiting, if not the performance desired, then at least something close to it. Either by constructing the artifact, or by simulating it, any **discrepancy** between desired and measured/predicted performance is noted. The designer then estimates, either from experience or, in some cases, by means of additional computation, those parameter changes that are likely to bring the performance closer to that which is desired; in other words, to reduce the discrepancy between desired and expected performance. Those parameter changes are then made, and their effect assessed, again by simulation or measurement. This procedure is repeated until the customer's specifications are satisfied or until it is decided that the proposed artifact is incapable of the required performance. Thus, the majority of engineering design carried out today is iterative in nature. More than one hundred iterations, and often many more, are typically involved.

2. Design technology

2.1. *Insight and visualization*

Graphical support that enables the designer to draw an artifact, to sketch, label and modify performance specifications, and also to compare them with predicted performance, can immensely enhance the design process because the designer can then *visualize* various aspects of the artifact (Baecker 1981, Fischer and Nieper-Lemke 1989). For the human designer who can naturally visualize in only two or three dimensions, computer-aided visualization can also provide insight into the complex multidimensional interrelationships among parameters and performances. Visualization tools designed to provide this insight can also acknowledge the advantages of being able to highlight important properties such as strong and/or orthogonal effects. The benefit of—indeed the necessity for—powerful visualization mechanisms is a theme which runs through this paper, where it is shown to be vital to other activities such as *automated* design.

2.2. *Automated design*

With the express intention of making the design process easier, much research has been devoted to the development of numerical techniques which, coupled with an ability to simulate the performance of an artifact, will adjust parameter values automatically and systematically in order to reduce—and if possible eliminate—the discrepancy between desired and actual performance. Such a technique is known as **automated design** or **optimization**, and has been proven extensively to be a powerful technique. Like manual design it is also iterative in nature.

One of the underlying themes of this paper is the proposition that, for the expected potential of automated design to be anything like fully exploited it must, paradoxically, be guided by the human designer. The pilot of an aircraft, for example, can call upon an autopilot to guide that aircraft to a specific destination but, nevertheless, can at any time observe progress and regain manual control when appropriate. In the same way the engineering designer must be able to specify the objective of the automated design but, likewise, be able to observe its progress and, for many different reasons, revert to the purely manual adjustment of parameters when necessary. To continue the analogy we may observe that, just as the aircraft cockpit contains many alarm systems actuated by an extensive monitoring of the aircraft's performance, the artifact designer could also benefit from alarms triggered by the extensive monitoring of the artifact's predicted properties as they change during automated or manual design.

Such human monitoring and control of automated design, as well as the response to alarms, requires, if it is to be effective, a supportive interface. In view of the close analogy between a pilot's use of an autopilot and the guidance of automated design by a human designer, the interface we have created for this purpose employs a Cockpit Metaphor, and is called the **Cockpit**.

2.3. *Expert advisory systems*

There is an obvious danger in offering the engineering designer a new tool or technique with whose fundamental operation they are unfamiliar. The danger is that of the burden of an additional skill to be learned over and above that of using a new interface. For this reason it is desirable to provide an expert system (Gupta and Rankin 1991) which will remove the need for that new technique (here, optimization) to be fully understood. Additionally, with the current state of simulators and knowledge-based systems, it is possible to provide a domain-based expert advisory system, offering the sort of advice that another skilled designer or technician might provide. In the design tool CoCo (Colgan *et al.* 1991),

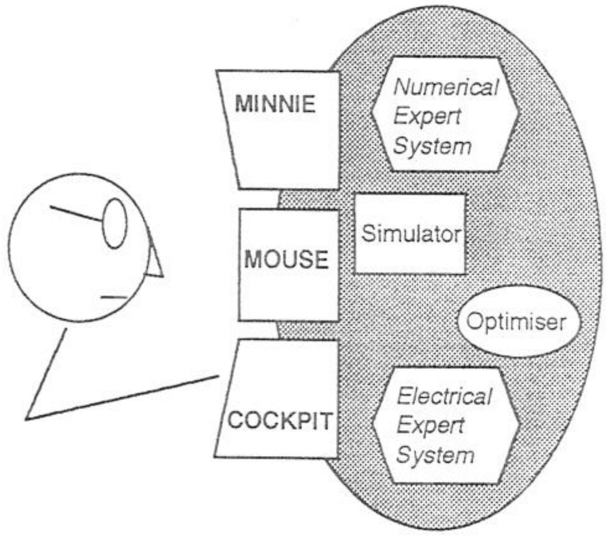


Figure 4. Principal components of the CoCo system for electronic circuit design.

now to be described, the Cockpit interface is additionally the channel for the provision of these two types of expert advice.

3. The CoCo design tool

3.1. *Electronic circuit design*

The potential design support offered by simulation, optimization, and expert advisory systems was recently demonstrated through the implementation of an experimental, but fully operational design tool for use by designers of electronic circuits. It was called CoCo, for the control and observation of circuit optimization that it facilitated. CoCo is regarded as a working statement depicting a profitable direction in which computer-aided circuit design systems may develop. Only two other experimental systems (Ketonen *et al.* 1988, Baker *et al.* 1989) are known to offer both optimization and expert advice, and none, to our knowledge, offers the type of novel interface which is the subject of this paper.

CoCo's functionality is indicated in figure 4 by the major modules of which it is composed. Performance specifications are described by means of an interface called MOUSE (Rankin and Siemensma 1989), while the description of the electronic circuit and the viewing of its predicted performance takes place via the MINNIE (Spence *et al.* 1986) interface. It is the Cockpit interface that permits the human guidance of automated design and communication between expert advisory systems and the designer.

Some modules comprising the CoCo system, such as MINNIE and the simulator (Philpac), have been in industrial use for almost ten years. New modules, including MOUSE (approximately 150K lines of code), the Cockpit (about 30K lines of code) and the optimizer and expert systems have been implemented to the same standard to permit real use by real designers. The entire CoCo system consists of about 500,000 lines of code, written in 5 languages.

Figure 5 shows the principal interface displays associated with the use of CoCo leading to the point at which optimization is invoked and the Cockpit is relevant. Figure 5(a) shows a circuit drawn using the MINNIE interface, while (b) shows the post-processed result of the simulation called by MINNIE. Figure 5(c) shows the result of using MOUSE to describe the specifications on AC circuit performance, and (d) shows how upper, lower, nominal and actual DC quantities can be displayed via a 'meter'. The way in which priorities associated with responses can be assigned is illustrated in figure 5(e): the length of each horizontal bar, representing a response, can be increased or decreased by means of the mouse, whereupon the lengths of the remaining bars will change to indicate their new relative weights. Figure 5(f) shows how the design variables are identified.

3.2. *Interface requirements*

The detailed nature of the Cockpit interface was determined by requirements derived from widely accepted views regarding the nature of electronic circuit design. However, the generic nature of engineering design—and, we suggest, of the Cockpit metaphor—is reflected in the fact that these requirements can be (and are, below) stated in general terms which are equally applicable to, say, structural and mechanical design. The requirements are labelled for later reference.

A As engineering design typically proceeds, concern and attention ranges continuously between the overall objective, the sub-objectives and more detailed artifact properties in a **hierarchy** of properties.

B Requirements upon a design can take three forms. In order of priority the first are **'hard constraints'** that simply *must* be satisfied: the fire resistance of building materials is one example. **'Soft constraints'** of the form 'make it as large as possible' are next in importance. Last in priority are **'principles of good practice'**. In an interface, and in a designer's mind, discrimination between these classes of requirement should be facilitated.

C In many stages of design the designer thinks **qualitatively**, ranking problems and design issues in order of importance or difficulty, but at any time may wish to check a numerical value.

D If an optimization is carried out in 'batch-mode', with only the final result examined, then the designer will lose any insight that may be gained from observation of the **progress** of optimization

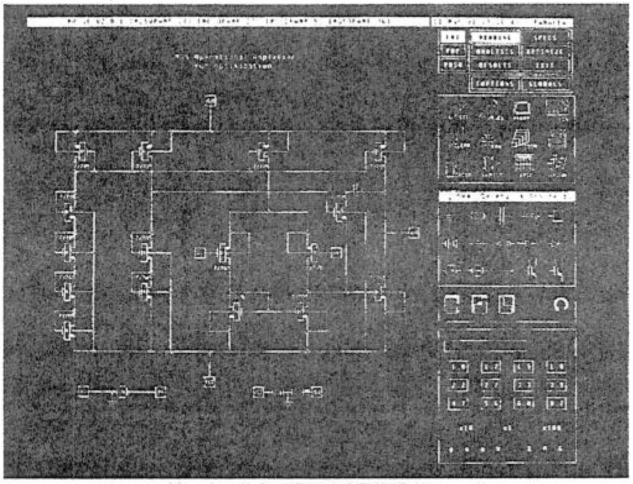


Figure 5 (a)

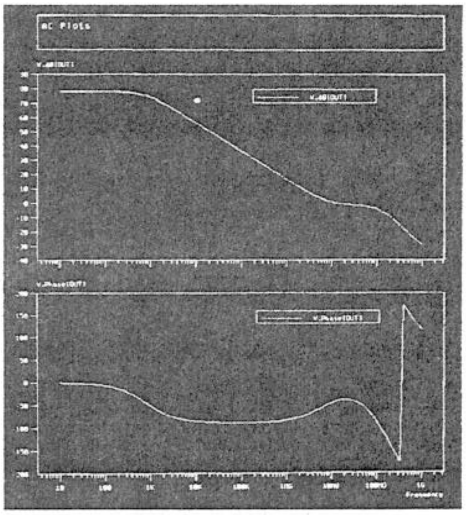


Figure 5 (b)

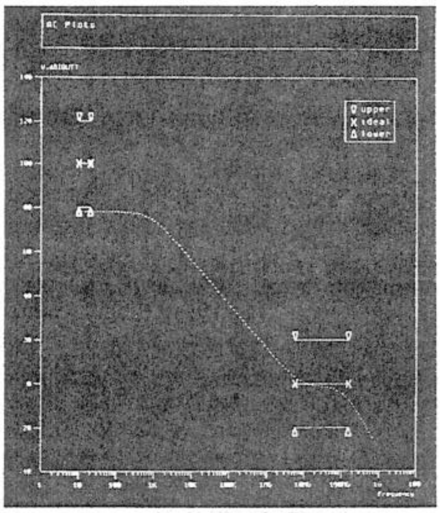


Figure 5 (c)

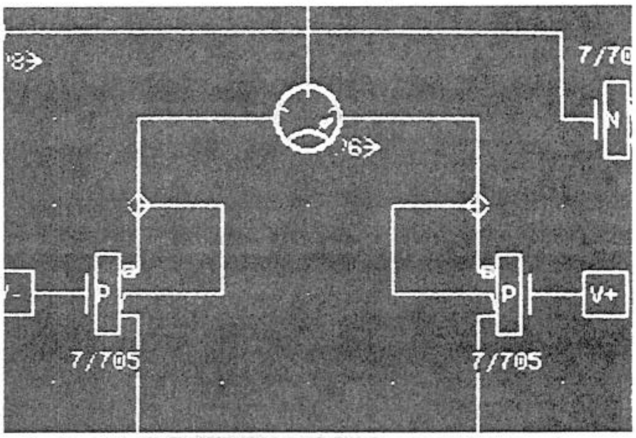
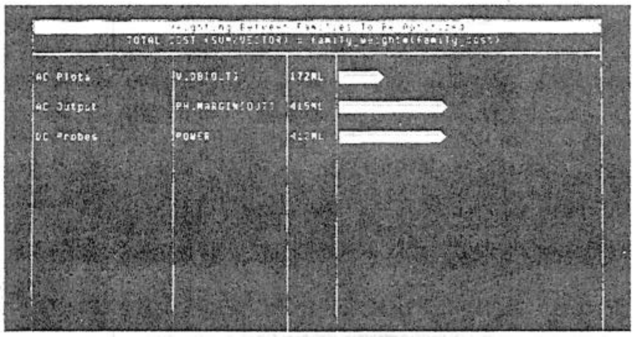


Figure 5 (d)

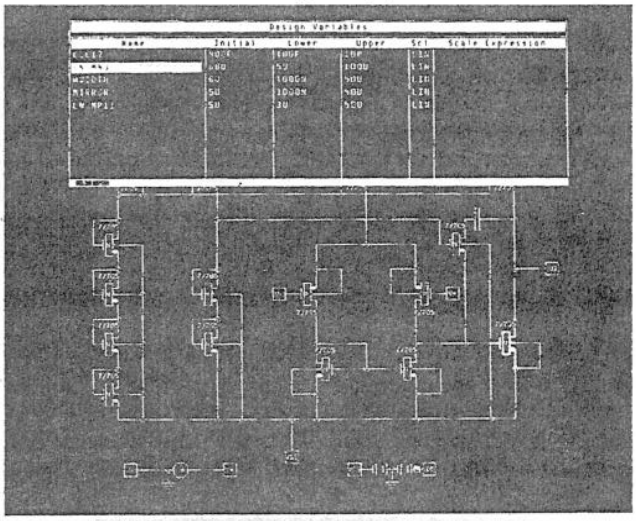


Weighting Between Families To Be Optimized

TOTAL COST (SUM/VECTOR) = family_weight*family_cost

AC Plots	V.DB(OUT)	172RL
AC Output	PH.MARGIN(OUT)	415RL
DC Probes	POWER	412RL

Figure 5 (e)



Design Variables

Name	Initial	Lower	Upper	Scl	Scale Expression
[illegible]	[illegible]	100F	10P	LIN	
[illegible]	600	5U	1000U	LIN	
[illegible]	60	1000N	500	LIN	
MIRROR	50	1000N	500	LIN	
LW MP11	50	30	500	LIN	

Figure 5 (f)

Figure 5. Examples of the principal interface displays involved in the use of CoCo up to the point at which optimization is invoked and the Cockpit interface is required to allow monitoring and guidance. Circuit drawing (a) is followed (b) by the display of simulated circuit performance, whereupon limits can be assigned both to (c) plots of AC performance and, (d) via 'meters', to DC performance. Bars whose length can be adjusted interactively (e) allow weights to be assigned to indicate the priority on performances, and tables (f) allow the designable variables (adjustable parameters) to be identified.

as it is affected by trade-offs and conflicts. In fact, the trajectory of the optimization can cover many alternative designs and trade-offs.

E It is extremely valuable for the designer to be aware of the **sensitivity** of properties of interest, including the overall objective, to changes in parameter values.

F If automated design is taking place, the designer would wish to be aware of the continuously changing parameter values, perhaps to see how close they are to practical manufacturing limits.

G At the start of a design the designer generally has little insight into the interrelations between important properties, e.g., the sub-objectives.

H In the past, designers have rarely been asked to define, mathematically, a single overall objective—that is, a measure for the overall 'quality' of the designed artifact.

I Designers tend to have geographic associations with their mental map of the artifact being designed. An example is the location of components on a silicon chip or printed circuit board.

J Designers cannot rely totally on automated design to alter parameter values. In particular, designers often wish to vary a parameter, not to improve performance, but rather to gain insight into the parameter's effect.

K At any point in the design process the designer may wish to suppress detail in order to concentrate upon global patterns.

L At all times the designer is hoping to gain insight into the complex interrelations among parameters and properties.

M On many occasions a designer is trying to locate the reason for residual sources of discrepancies, or unreasonable performance specifications.

N The designer welcomes advice concerning both the choice of an appropriate optimization algorithm and the existence of potentially serious conditions arising during the design.

Preliminary designs of the Cockpit were prototyped and tested on real designers, and most of the above requirements were supported by the final design of the Cockpit.

4. The Cockpit

An overall view of the Cockpit is provided in figure 6 (colour plate). Below we discuss the four windows within the Cockpit, with references such as '(**D**)' to indicate the relevant itemised requirement listed in the previous section.

In the main window is a pseudo-3D presentation of a tree-structure comprising linked sheets, to mirror the strong sense of hierarchy in engineering design (**A**). Figure 7 shows, diagrammatically and in perspective, the topmost sheet, which is associated with the single overall objective. On this sheet the size of the (red) circle indicates the discrepancy between the desired and achieved values of the objective. As optimization proceeds, the size of this circle reduces, ideally becoming zero. The use of circles to encode data qualitatively (**C**) is not new, having been promoted by Sir Edward Playfair in 1801 (Tufte 1983) and later exploited in one of the earliest interactive-graphic engineering design systems (Spence and Apperley 1977).

The overall objective is the weighted sum of sub-objectives whose discrepancies are similarly represented by circles in the second row of sheets. These, in turn, could well be related to different properties further down the hierarchy: it would not be unusual for 6 levels of sheets to be involved in a design. As figure 8 shows, dependence within the hierarchy is made overt, and simple controls on each sheet allow a subordinate sheet to be unfolded or suppressed (**K**). A scroll bar is useful when the height of the hierarchy exceeds that of the main window. The hierarchical display is especially valuable when the designer is trying to track down residual sources of discrepancy (**M**). As optimization proceeds, correlated changes in circle sizes—for example with a large circle deflating and, simultaneously, a small circle inflating—will often indicate trade-offs (**D**) or similarly important relations, and thereby provide insight (**L**) into the artifact's design. If necessary the location of circles on a sheet can correspond to the location of electrical components on a board or chip (**I**).

If more detail regarding a discrepancy than that provided by circle size is required (**C**), an 'orientation button' on each sheet allows it to be presented in a 'plan view' (figure 9) displaying the numerical value(s). As well as providing these two representational modes, a pull-down menu (figure 10) additionally allows the presentation either of current state, a history (**D**) of the properties as they change during optimization, or progress (**D**) in the reduction of a discrepancy.

In the **Design Variables** (parameters) window (figure 6 centre, bottom) parameter values are represented qualitatively (**C**) by the upper row of bars: a full bar indicates a parameter at its upper limit, an empty one at its lower limit (**E**). As optimization proceeds, the bars change accordingly, and superimposed chevrons indicate the direction and rate of change. The layout of parameter bars and property circles is seen to mirror figure 1. The lower row of bars, together with '+' and '−' signs immediately below, similarly indicate the sensitivity (**E**) of the overall objective (represented by the single circle on the top sheet) to changes in parameter values. As with the discrepancy circles in the main window, quantitative (**C**) and history information (**D**) concerning parameters is also available via pull-down menus.

The **video control** window (figure 6, bottom left) contains

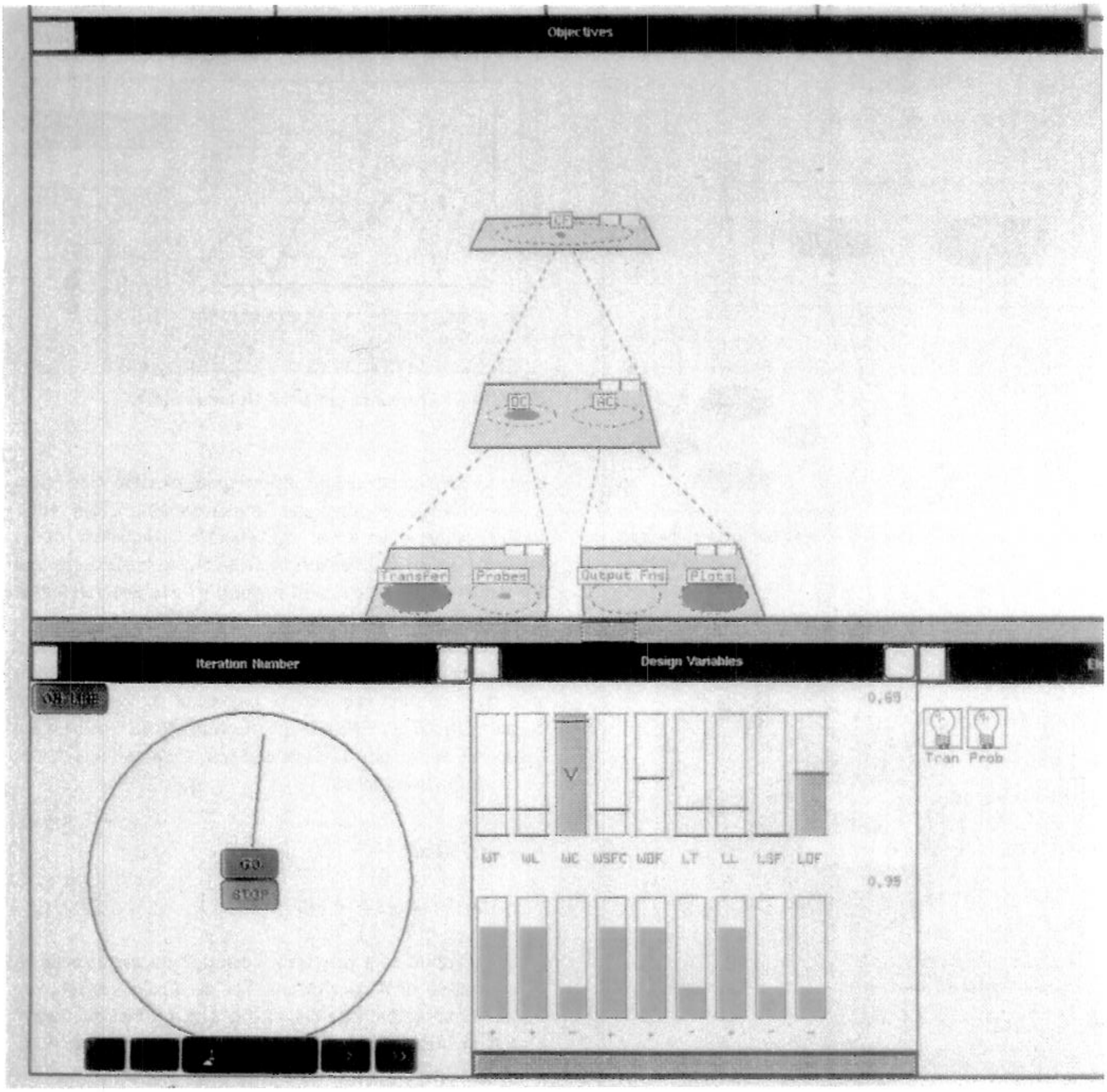


Figure 6. The Cockpit interface.

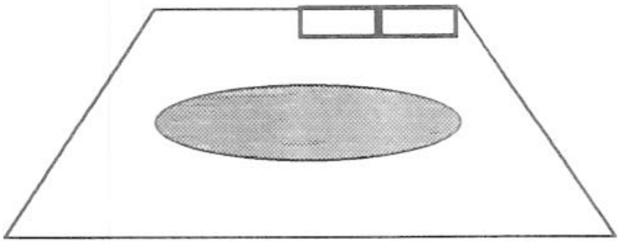


Figure 7. The size of the circle on the topmost sheet indicates the discrepancy in the objective function.

a clock which indicates the optimization's iteration count as well as controls which allow the (recorded) optimization results to be replayed, perhaps to enable the correlated changes of two circles to be examined in detail (**D**).

The **alarms** window (figure 6, bottom right) is used to alert the designer to potentially serious electrical conditions (**N**). As optimization proceeds, conditions within the current circuit are monitored automatically, and a knowledge-based subsystem identifies potential problems. These are signalled to the designer by a change in the colour of an icon. Such an

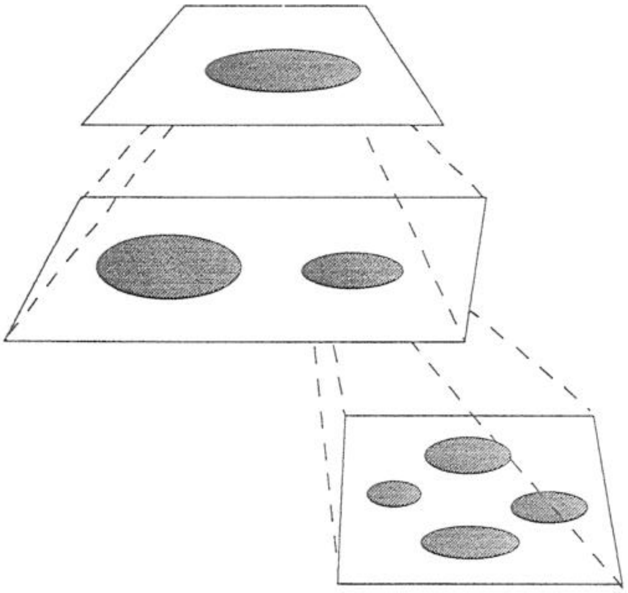


Figure 8. A hierarchy of sheets, reflecting the hierarchical nature of engineering design.

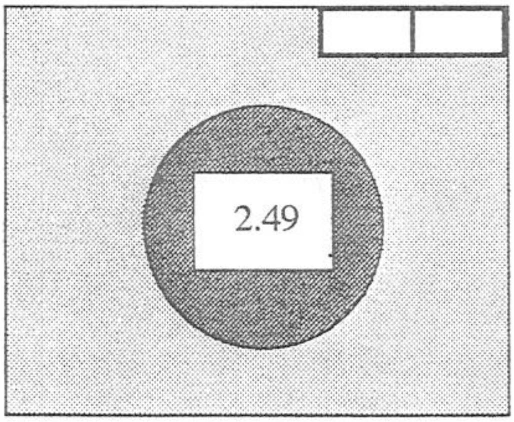


Figure 9. The plan view of a sheet.

alarm will often (but not necessarily) be followed by interrogation by the designer, leading to the display of a simple message and, if required, an identification on the circuit diagram displayed in the MINNIE interface as to where the problem has arisen (Gupta and Rankin 1991).

5. Contribution

An important characteristic both of the CoCo system as a whole and the Cockpit in particular is that they were *not* designed primarily in response to customer requests. They are the implementation, by domain experts and a psychologist, of long-term projections about the needs of the circuit designer. We are, therefore, reporting **innovation** rather than a commissioned design and implementation. This innovation, together with the considerable complexity of its context (CoCo) and the fact that CoCo is a real design tool for industrial problems, will hopefully have generated new techniques and concepts relevant to the field of human-computer interaction. Thus, in what now follows, we are saying, with respect to both CoCo and its Cockpit component, 'Here is a new concept we feel is important to the future of engineering design; it has been observed in real use; here are the interesting issues it raises; and here are possible avenues of future improvements.'

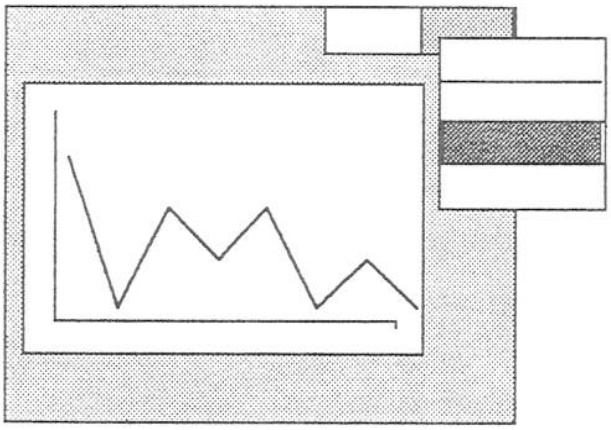


Figure 10. Alternative presentations are available.

6. Evaluation

6.1. *Major factors*

The creation of a new tool is naturally accompanied by consideration of its evaluation. For the CoCo system, two major factors influenced the way in which it was evaluated. One is the fact that, due to the inherent novelty of the tool, there was no 'control' easily available for comparison. Thus, no attempt was made to compare the use made of CoCo with that made of any other existing system. It was felt to be far more useful to the advancement of engineering tool design to study the use, utility and characteristics of CoCo with a view to identifying ways in which its value to the designer could be enhanced. It is for this reason that studies of CoCo (undertaken before the Cockpit was complete), fully reported elsewhere (Colgan *et al.* 1991), concentrated on actual use by real designers and protocols appropriate to the testing procedure. One of its results, for example, was a cognitive model of design which may well help to guide the future development of CoCo and similar systems (Colgan and Goebel 1993). While these tests without a 'control' may be disappointingly informal from the view of scientific cognitive experimentation, they reflect a practical approach to innovating and improving an industrial design tool (Carroll and Campbell 1986).

The second factor influencing the conduct of evaluation was the very high cost of studying its use in a controlled manner. A significant aspect of the CoCo system is that it is a *real* industrial tool for use by *real* designers creating *real* products, and initially it was only from the observation of realistic use that we considered it possible to draw conclusions regarding potential improvements and extensions. However, to be of any value, systematic observation of the use of CoCo by just *one* industrial designer costs around $5000. In view of this, time and economic considerations permitted the study of only three designers. Two were used in the earlier CoCo system study (Colgan *et al.* 1991), and one (see below) in the controlled study of the Cockpit interface.

6.2. *Evaluation of the Cockpit*

Evaluation of the Cockpit has proceeded along three lines.

First, in actual use, we have already observed the Cockpit supporting *diagnosis*, as when a failure of the top circle to decrease further was traced down the hierarchy to the underlying cause, and *discovery*, when rapid correlated changes in the sizes of two circles drew attention to an important trade-off relation. In both these cases a cognitive problem was transformed into a much easier perceptual problem.

Next, in order to identify potential improvements in the Cockpit interface, a more formal evaluation (Colgan 1991) involving *real* use by a *real* designer was undertaken. Only one subject was involved (a very experienced and skilled circuit designer) in view of the considerable expense. A substantial setting-up procedure, involving 10 person days, was needed for the provision of correct device models and simulation capability to ensure that no unexpected situations detracted from the Cockpit evaluation *per se*. The actual experiment involved the subject for three half-days, first in training, then in a circuit design involving both optimization and the electrical expert system, and then the completion of a questionnaire and a debriefing. Analysis by the experimenter was spread over a period of fifteen days. The designed circuit (Colgan 1991) was a CMOS operational amplifier, a circuit considered to be medium in complexity: it contained 4 designable parameters and 4 performance specifications.

6.3. *Outline of the Cockpit evaluation*

During the design session, the designer set up the circuit in MINNIE and MOUSE, simulated and viewed the circuit's performance and set up an optimization profile. The optimization profile consisted of specifications for the ideal performance of the circuit, the relative weightings of performances and a selection of the design variables to be adjusted. Electrical alarms were associated with selected components and clusters of components. The designer then used the Cockpit to view the results of the optimization.

Analysis of the design process adopted during the trial resulted in a series of glimpses of how the Cockpit may influence design behaviour. From the previous studies of design without the cockpit interface, design behaviour was categorized into several design strategies. We shall not review them here, as this has been done elsewhere (Colgan *et al.* 1991), but many strategies were concerned with problem simplification. For example, *problem decomposition*, which allows design goals to be decomposed into their constituent sub-goals; *chunking*, which permits less important details to be ignored in the process of recognizing meaningful patterns; *dimension reduction*, which results in complex interdependencies between designable variables being ignored whilst two or three are varied; *identification of weakly interacting sub-modules* to allow each sub-module to be designed separately; and *estimation of performances* to save valuable design time. All these strategies involve approximation of some part of the design.

These strategies were much less noticeable during the use of the Cockpit. It could be argued that, in fact, the Cockpit has already naturally removed some of the detail for the designer by presenting a high-level symbolic representation of the design, thus rendering the (previously explicit) design simplification techniques redundant. The designer continued to view the design process at this high level for much of the design review, and used this view to explore the results of each iteration to find the 'best' iteration which, interestingly, was not the same as the optimiser's last iteration.

The designer spent much less time exploring possible trade-offs which previously (without the Cockpit) had been explored through the process of hypothesis and test (where the test was by simulation). Trade-offs were more directly observable from the Cockpit. The designer used a process of moving backwards and forwards through the iterations to locate trade-offs *visually*, as evidenced by the relationship between the sizes of two ellipses on a sheet. The designer also cycled manually through the iterations in order to follow the changes made in the design variables. Previously the strategy of dimension reduction had been used to reduce the design variables to a manageable number for optimization, but the graphical displays of the Cockpit allowed the effect of design variables on performances to be traced as the iterations are incremented.

6.4. *Conclusions*

The main outcome of this exercise was evidence that the

Cockpit metaphor was useful in real design. The designer viewed the interactivity provided by the Cockpit during optimization as an essential, informative feature, especially at the 'mid-design' stage in order to obtain some 'global feel' for the relation between design variables and circuit performance. During a short tutorial, the subject easily grasped the main concepts of the Cockpit's symbolic display, appreciated the qualitative display of both optimization results and design variable values, and expressed a preference for the symbolic presentation of information rather than standard graph displays. The subject was able to identify three sets of significant trade-offs in the design, but suggested that their presentation be given more emphasis and be more obvious.

The subject felt it was useful to be able to interrupt the optimization to see what progress was being made, but suggested that it should additionally be possible to adjust parameters *manually* in order to increase the extent of manual exploration available within the Cockpit environment: adjusting design variable values at various stages during the optimization, and viewing the results in the Cockpit, would provide vast insight into circuit behaviour. Nevertheless, in rating his confidence in having chosen the correct set of design variables, the subject gave only a medium response, indicating that he was prepared to believe that there were other design variables that would have a stronger effect during optimization. Indeed, it was remarked that it would be useful if CoCo itself could suggest 'strong' design variables. Both of these points—manual interaction and the identification of strong variables—could be addressed by extensions within the Cockpit metaphor and adaptations to the underlying CoCo system.

Low level issues, such as the appropriateness of the scaling algorithm for ellipse size, were uncovered. The subject also indicated that he was not comfortable with terms that were mathematical in nature (e.g., 'objective function'). Finally, the text associated with the electrical alarms was considered to be insufficiently informative: as well as being told that a particular transistor had moved into a different operating mode, the subject wished to know exactly *why* this had occurred. Again, these points could be addressed by adaptations to the system.

A third and more informal evaluation of the Cockpit has independently been carried out in industry (Hoeksma 1992) by tool designers considering the extension of the CoCo system to support, in addition to optimization, the more complex tasks of design for manufacturability (Spence *et al.* 1988) and design for reliability (Brombacher 1991). For example, rules for reliability would be set up in a knowledge base (Gupta and Rankin 1991) which, in turn, would trigger alarms in the appropriate window of the Cockpit. As with optimization, it is again anticipated that the Cockpit will allow designers unfamiliar with the underlying mathematical and modelling techniques (Brombacher 1991) to observe and control the redesign of an electronic product for improved reliability.

7. Critique

The Cockpit, as illustrated in figure 6, has additionally and inevitably been under continuous informal evaluation by its designers for more than a year. Many useful design freedoms and potentially desirable changes have been identified.

One such change refers to the priority ordering of hard constraints, soft constraints and alarms, referred to earlier (**B**). Such ordering can usefully be reflected on the sheets as shown in figure 11. A hard constraint takes precedence, and its violation is indicated by a square (rather than a circle) both on the sheet on which it occurs *and at all parent levels*, the circles on these sheets being suppressed. A hard constraint violation can then be traced down the hierarchy to see where the problem has arisen. Alarms, which constitute 'advice' open to discretionary action by the designer are indicated only on the sheet where they occur, and do not cause any suppression. Thus, the interface filters the presentation of large quantities of results in a way which is natural in that it reflects the priorities in the designer's mind. To permit manual design and exploration, it would also be useful if the parameter bars could be made sensitive to manual interaction.

Other design freedoms which are available, and which are under active consideration, include the 'depth' and colour of the discrepancy circles, their partitioning to indicate their sensitivity to parameter variation and a more direct indication, on the sheets, of performance specifications. Because, in conventional design, designers do not normally think explicitly in terms of a single objective function (see **H**), and might well have difficulty in choosing weights, the desirability of displaying the top (objective) sheet is also placed in

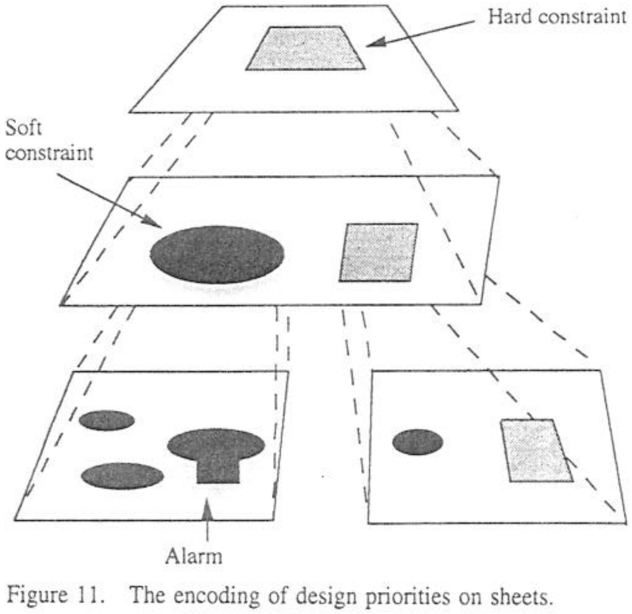


Figure 11. The encoding of design priorities on sheets.

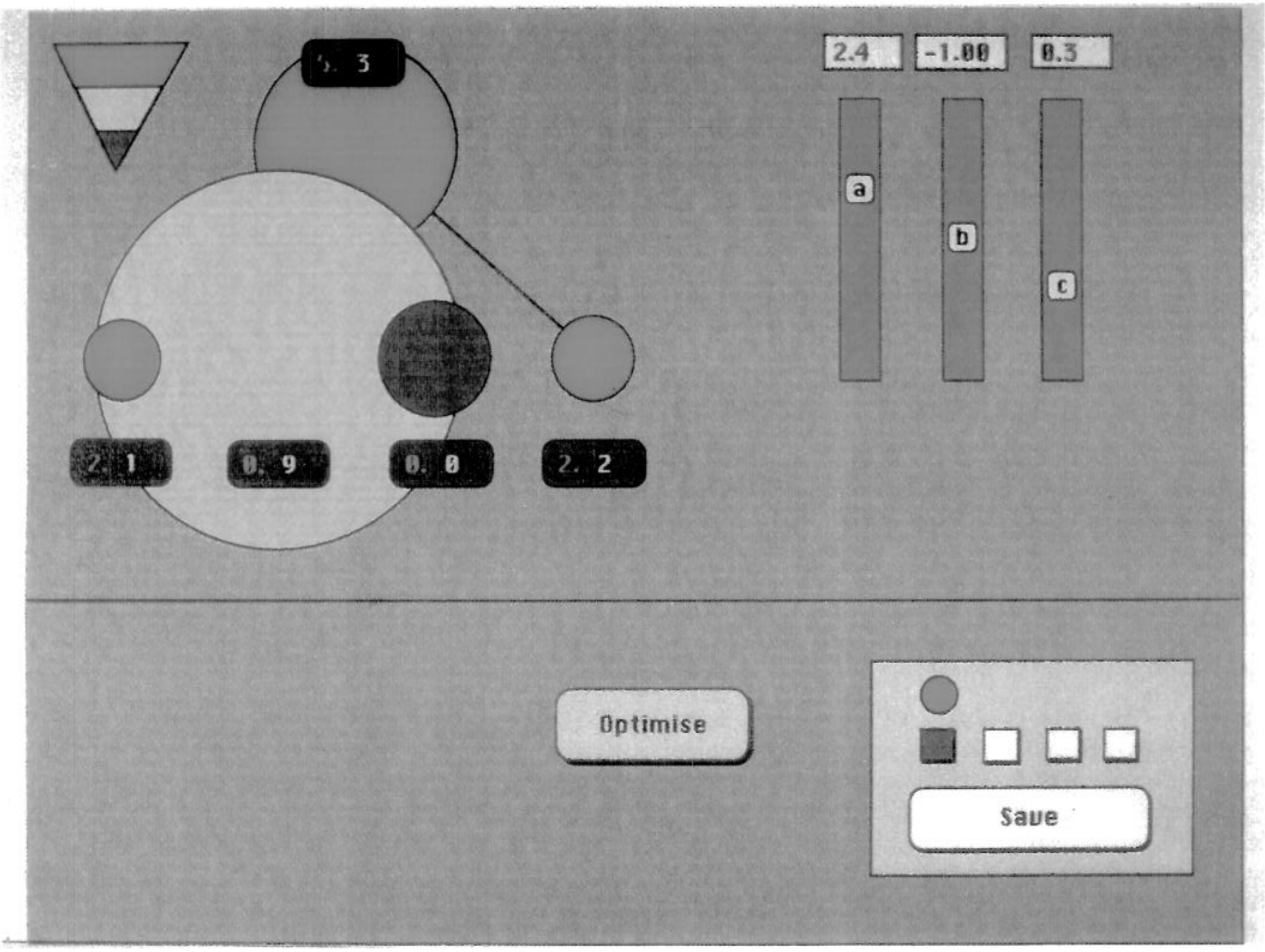


Figure 12. The mini-Cockpit.

question. Finally, since the main wondow of the Cockpit is of finite width, some technique such as scrolling or the use of a bifocal display may have to be offered to cater for the situation in which more than two or three sheets are displayed at the same level.

8. Future extensions

Interesting avenues of research have been identified by our experience with the Cockpit, and many are directly relevant to the constant need to improve design tools. The following examples provide illustrations of three such avenues of research.

8.1. *The mini-Cockpit*

There is a need to investigate, at reasonable cost, the nature of the dialogue between the designer and the Cockpit. To this end a laboratory-based 'mini-Cockpit', of which the current version (Toker 1992) is shown in figure 12 (colour plate), has been developed. To support manual as well as guided automated design, scroll bars (of which three are shown in figure 12) allow parameter values to be adjusted by hand (**J**) as well as their automatic adjustment observed. To reduce complexity a 2-level performance hierarchy is provided, and a wide range of discrepancy values, encoded as circular areas, is handled by the use of colour to denote scale, a constant reminder of which is provided by the triangle (at upper left of figure 12). To study the way in which designers may 'tag' promising candidate designs and return to examine them later, a 'record box' provides both retrieval selector buttons and, above each, an indication of the associated objective (discrepancy) value.

8.2. *Temperature of exploration*

The second example concerns the possible value of new concepts to characterize the manner in which a designer explores multi-dimensional parameter space.

During some earlier (*c.* 1978) unpublished experiments, it was observed that, when given the ability to adjust two parameters in order to cause a property to lie within stated bounds (a task similar to that of designing an electrical filter),

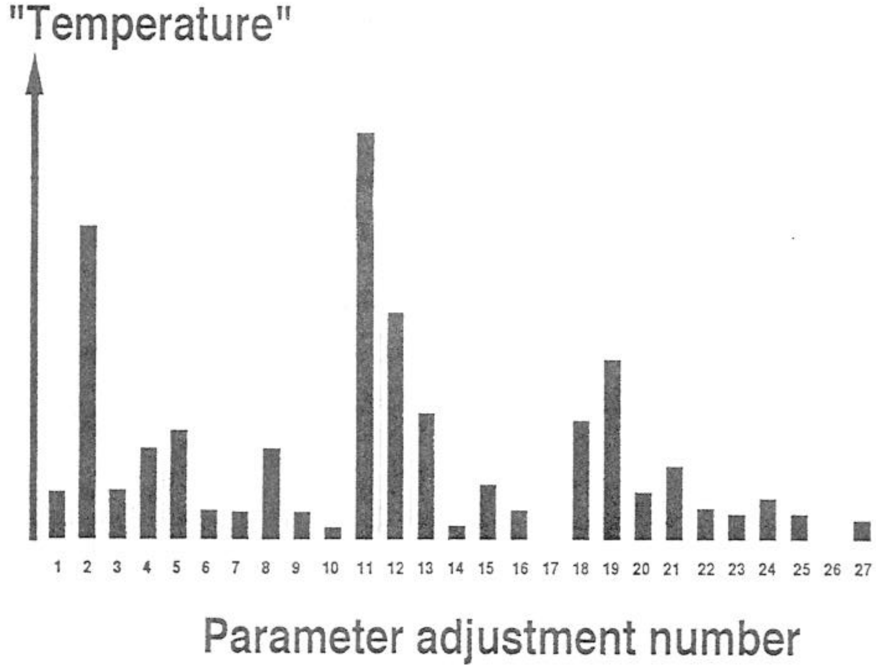


Figure 13. The 'temperature' of exploration of parameter space during a manual design.

some subjects exhibited a 'cautious' approach consisting of many small steps, with the consequent danger of achieving a local rather than a global optimum. In distinct contrast, other subjects began with a 'global exploration' consisting of large and apparently quite random steps, with the attendant chance of locating the vicinity of a global optimum. Mindful of the relation between the energy of atoms and temperature, it was suggested that the concept of 'temperature' might be used to characterize the nature of the exploratory manual adjustments often undertaken by an engineering designer. The idea was briefly tested using the mini-Cockpit shown in figure 12.

The result (Toker 1992) is shown in figure 13. For each of 27 successive parameter adjustments, the 'temperature' was calculated as the exploration range of a parameter divided by the time spent on that exploration. After the tenth parameter exploration the subject was informed that a global minimum had not been achieved, whereupon a similar pattern of exploration was repeated. The same situation occurred at the seventeenth exploration. Despite obvious fluctuations, as well as limited evidence, we speculate that the gradual decrease in temperature within successive explorations is sufficiently suggestive of the gradual transition between global and local exploration encountered in engineering design as to be worth further investigation.

8.3. *Qualitative design*

In the stage of design preceding that to which CoCo is suited, the designer is less concerned with precise parameter and performance values and much more with *which* parameter should be adjusted, the *general extent* of such an adjustment, and the *approximate* effect on circuit performance. This stage, in which the designer is thinking primarily in a qualitative mode, as been referred to as Qualitative Design. If qualitative design is best supported by the qualitative display of data, then many of the ideas incorporated in the Cockpit may be relevant to a tool designed to support this phase of design.

9. Conclusions

Notwithstanding detailed improvements that have been identified as desirable, many indicators point to the Cockpit metaphor as a way of exploiting and enhancing the value of automated design: it allows the designer to visualize an artifact's properties, to guide its design, to monitor alarms and to make engineering judgements more effectively. It is relevant to future CAD systems and to the convergence of process control monitoring and CAD interfaces. The metaphor has also raised, for HCI researchers, many issues of interest having direct relevance to the advancement of engineering design, and it has triggered further innovation and development in the field of engineering design tools.

Acknowledgements

The CoCo research was funded by Philips Research and the UK Medical Research Council. In addition to the authors, the interdisciplinary effort involved Ashok Gupta and Maddy Brouwer-Janse. The Cockpit metaphor's use for yield and reliability design has been assessed by Aarnout Brombacher and Ronald Hoeksma. Valuable suggestions were made by Mark Apperley and an anonymous referee.

References

BAECKER, R. 1981, Sorting out sorting, 30 min colour/sound videotape, distributed by Morgan Kaufmann.

BAKER, K. D., BALL, L. J., CULVERHOUSE, P. F., DENNIS, I., EVANS, J. ST. B. T., JAGODZINSKI, A. P., PEARCE, P. D., SCOTHERN, D. G. C. and VENNER, G. M. 1989, A psychologically-based intelligent design aid, *Eurographics Workshop in Intelligent CAD*.

BROMBACHER, A. C. 1991, *Reliability by Design* (Wiley, Chichester).

CARROLL, J. M. and CAMPBELL, R. L. 1986, Softening up hard science: reply to Newell and Card, *Human–Computer Interaction*, **2**, 227–249.

COLGAN, L. 1991, Human guidance of automated design, PhD thesis, University of London.

COLGAN, L. and SPENCE, R. 1991, Cognitive modelling of electronic design, in J. Gero (ed.), *Artificial Intelligence in Design* (Butterworth–Heinemann, Oxford), 543–559.

COLGAN, L. and GOBEL, M. 1993, Towards a cognitive model of circuit design, *International Journal of Environment and Planning. B: Planning and Design*, special edition on Cognitive Models of Design, June, **3**, 321–332.

COLGAN, L., RANKIN, P. and SPENCE, R. 1991, Steering automated design, in J. Gero (ed.), *Artificial Intelligence in Design* (Butterworth–Heinemann, Oxford), 211–230.

FISCHER, G. and NIEPER-LEMKE, H. 1989, HELGON: extending the retrieval by reformulation paradigm, *Proceedings of CHI'89*, (ACM, New York) 357–362.

GUPTA, A. and RANKIN, P. J. 1991, Knowledgeable assistants in design optimization, in J. Gero (ed.), *Artificial Intelligence in Design* (Butterworth–Heinemann, Oxford), 623–642.

HOEKSMA, R., private communication.

KETONEN, T., LOUNAMAA, P. and NURMINEN, J. K. (1988), An electronic design CAD system combining knowledge-based techniques with optimization and simulation, in J. Gero (ed.), *Artificial Intelligence in Engineering Design* (Amsterdam, Elsevier).

RANKIN, P. J. and SIEMENSMA, J. M. 1989, Analogue circuit optimisation in a graphical environment, *Proceedings of ICCAD-89* (IEEE, New York), 372–375.

SPENCE, R. and APPERLEY, M. D. 1977, The interactive-graphic man–computer dialogue in computer-aided circuit design, *IEEE Trans. on Circuits and Systems*, **CAS-24**, 2, 49–61.

SPENCE, R. and SOIN, R. S. 1988, *Tolerance Design of Electronic Circuits* (Addision–Wesley, Wokingham).

SPENCE, R., CHEUNG, P. and JENNINGS, P. 1986, MINNIE: the way ahead for analogue CAD, *Silicon Design*, March, 6–8.

TOKER, S. 1992, Design theory, testing, and optimization of the mini-Cockpit interface, Final Project Report, Department of Electrical Engineering, Imperial College, London.

TUFTE, E. R. 1983, *The Visual Display of Quantitative Information* (Graphics Press, Cheshire, Conn.).