Chapter 7
Complexity

Chris Earl and Jeffrey Johnson
The Open University

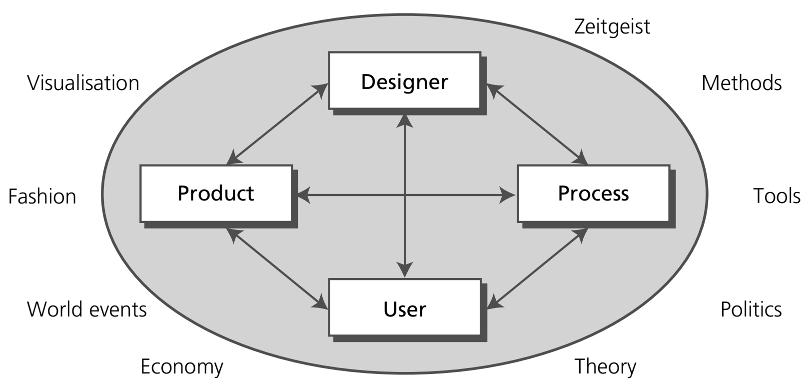
Claudia Eckert
University of Cambridge

Complexity is a widely used term; it has many formal and informal meanings. The aim of this chapter is to examine the relation between complexity and design. Several formal models of complexity can be applied to designs and design processes. This argument runs in two ways.

First, designing provides insights into how to respond to complex systems – how to manage, plan and control them. Second, the overwhelming complexity of many design projects leads us to examine how better understanding of complexity theory can lead to improved designs and processes. This second direction is the focus of this chapter. We start with observations on where complexity arises in design, followed by an overview of the scientific background to complexity to introduce the wider context in which the concepts and methods of complexity theory have arisen.

Many involved with design recognise some area of their work as complex. Figure 7.1 shows the main sites for complexity in design and designing. First, the product/service/system under construction may be complex in its own right, in structure as well as behaviour in use. Second, the process of designing may contain many interrelated tasks, each having many subtasks. Third, the designer and their part in the organisation of project teams integrating complex sets of capabilities and experience. Fourth, users, and those more widely affected through life-cycle effects such as environmental impacts, provide a complex context for designs.

The relationships between designs (products, services or systems), processes, designers and users create yet another level of complexity. For example, the relation between design and users includes the difficult and complex problems of sustainability – the widespread impacts of a design across populations and into the future. Figure 7.1 also indicates the wider context of designing which forms another level of complexity.



7.1 Designing in context

The relation between product and process is critical and is frequently the source of complexity. For example, scheduling the product across available design resources and capabilities which make up the process is a difficult task, not least because individual design activities in the process have uncertain durations.

The way that a product 'flows' across the resources and capabilities in the design process, with associated interactions between parts of the process, is complex. Managing these flows is a challenging task. As a design develops (through process) it is represented in several different ways. These representations and models may be complex in their own right. They may also be used in complex ways. Representations change in type and content as design proceeds from concepts and sketches to computational models and prototypes.

Designing can certainly be complex in the informal senses in which it has been described above. These observed characteristics are mirrored by established formal models and ideas in the science of complexity. In Figure 7.3 we summarise briefly the main points of complexity theory (see Suh (1999) for a brief summary in relation to design). These models have evolved to describe particular systems and their properties, which accounts for some of their differences. Many complex systems display aspects of several of these views simultaneously.

7.2 The EH101, complete with five composite rotor blades
© AgustaWestland

There is one additional point we would like to make. The way that designing develops intention, through concept to final design, appears to be an exemplar of how to model a complex system by increasing detail in representations through a process of iterative evaluation. Indeed, there may be lessons for complexity science itself from analysis of the way that design is undertaken (Cross, 2000), especially recent work on comparing processes across different domains (Eckert *et al*., 2004). We talk intuitively about complexity in design and know that it can cause problems. But can we understand and manage complexity in the different areas and levels of design? To answer this we do three things. First, we distinguish different kinds of complexity that are present in design. Second, we discuss the methods and techniques from complexity theory. Third, we seek to apply these to designing.

Complexity in an engineering context

A helicopter rotor blade is complex not only in its form and manufacture, but also in its functions. Its design process is complex to the extent that it eludes conventional process modelling, with a large number of closely

Differential and difference equation models represent
- Relations among variables describing the state of a system
- How state variables change with time
- Parameters which identify specific relations among variables
- Behaviour as described by solution trajectories → *system order* in equations and *behaviour uncertainty* in trajectories

Types of system
- Conservative systems (respond to perturbations with permanently altered behaviour)
- Dissipative systems *(absorb* perturbations, returning to a *steady state* behaviour) (Nicolis and Prigogine, 1989)

Types of behaviour
- Lyapunov stable – behaviour changes proportional to perturbation, e.g. planetary orbits
- Asymptotically stable – behaviour returns to steady state (an attractor) after a perturbation
- Unstable behaviour departs radically from the initial state
- Locally stable (below a threshold in perturbations)

Non-linear equations
- Combined effect on behaviour of perturbations (with small effects individually), is non-linear (superposition does not apply)
- Behaviour may be unstable and difficult to predict

Chaotic systems
- Different types of trajectory which are *very close* to one another at certain parameter values
- Small unmeasurable disturbances alter system parameters knocking the system from a stable to an unstable trajectory, or from an unstable to an asymptotically stable trajectory (a chaotic attractor)
- Behaviour cannot be predicted because of inherent measurement uncertainty
- Chaotic behaviour in one element can propagate across the entire design
- Designed systems may potentially chaotic, e.g. aerodynamic and road systems often perform best with parameters *on the edge of chaos*

Information measures of complexity
- Expected information (Jaynes, 1957) or algorithmic complexity (Chaitin, 1987)
- Balance system order and behaviour uncertainty

Synthetic systems models
- Rules and goals indicate order
- Simulation reveals uncertainty in behaviour

Nearly decomposable systems
- Strong relations within parts and weak relations between parts (Simon, 1969)
- Techniques for identifying near decomposability are widely used in models of design process (Eppinger *et al*., 1994; Suh, 2001)

Fractals and cellular automata
- Simple rules generate complexity, e.g. fractals (Mandelbrot, 1983) and cellular automata (Wolfram, 2002)
- Applications, e.g. urban development (Batty and Longley, 1994; Wilson, 2000)

Adaptation and coevolution
- Adaptation – change behaviour in respose to environment
- Coevolution – mutual adaptation, e.g. simulation of both transportation infrastructure and land use (Barrett *et al*., 2001)

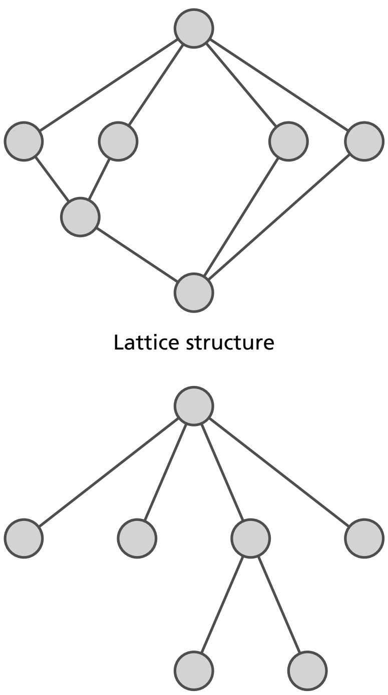
7.3 Overviews of the theoretical models of complexity

interdependent and related shape and material parameters which are determined iteratively (Clarkson and Hamilton, 2000). Off-road diesel engine designs are customised for users and subject to environmental impact legislation. Their complexity lies in the interactions between product and users (and the logistical effort involved in designing and producing thousands of slightly different products). Power generation switchgears are customisations of standard products completed on a contract basis. Managing several different products through the design and manufacture process produces complex scheduling problems under constraints of uncertainty and finite capacity resources (Earl *et al.*, 2001).

Product structure

A design may be structurally complex – an engine has many parts and specific functional relations between parts. Parts and relations between parts form a hierarchical structure which is not necessarily tree-like but may display more connected lattice properties (Figure 7.4). A bill of materials (BOM) for manufacture describes the structure of a product in terms of which parts are included in aggregate units. A BOM can go to the finest detail of components and is in the form of a tree-like 'explosion' of the product.

For a product with many components, the BOM may be a broad and deep structure with main parts having many subparts (breadth) and these in turn being decomposed repeatedly until the final manufactured components are reached. Companies can reduce the breadth and depth of BOMs trees by taking delivery of whole subsystems from suppliers. However, the BOM structure, although complicated, is not really complex. It has been handled by materials and manufacturing planning software which has proved an invaluable basis for manufacturing planning generally. A product has other structures associated with it during its development and it is the interaction among these structures which presents the complexity designers experience in product development.



Lattice structure

Tree structure

7.4 Tree and lattice structures

Product structure is a decomposition which corresponds to functional parts of a design. Parts at one level of the decomposition may 'belong' to several larger functional parts. Thus, a rotor shaft in a jet engine 'belongs' to both the turbine and the compressor. The shaft itself has two parts, one for the turbine rotor and another for the compressor rotor. This kind of relationship among parts is not captured by a tree-like hierarchy, but requires a lattice hierarchy.

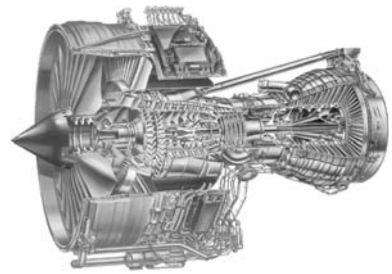
In these lattice structures, for any two parts there is a unique ('smallest') part at a higher level of the decomposition to which the two parts 'belong'. In the case of the rotor shaft, the two parts belong to the part 'rotor shaft' as well as to the turbine and compressor. These are functional parts of the whole jet engine, as is the rotor shaft. The 'smallest' higher level part to which both parts of the shaft belong is the rotor shaft. Notice that the decomposition of the engine we are describing here is not one which divides the design into distinct parts; there is considerable overlapping.

There are further descriptions. The manufacture and assembly of a product is described in terms of the precedences and sequences of operations. This structure may be quite different to the product structure as the simple example of the rotor shaft shows.

These different hierarchical descriptions may not be particularly complex in themselves. However, with several different descriptions used by different teams during product development, the result can be very complex. This is compounded by the nature of the design process in which descriptions are constantly changing as details of concept are completed, suppliers contracted and manufacturing planned. These structures are central to understanding complexity in design and are reviewed in greater detail later in the chapter.

The BOM illustrates many of the problems of describing a complex product. For manufacturing and assembly a BOM is fairly unproblematic. It indicates which parts are assembled together and it is used to track parts. Every part has a unique place in the BOM – it is a tree structure. However, in many companies a BOM is a problematic concept in design. Designers are interested in systems and their parts. The BOM is used to track progress in design, in terms of what percentage of the BOM has already been designed.

From a design point of view, conflicts can arise when several people work on the same part independently, or when nobody does. Important subsystems can easily be buried in a BOM, either because the parts are distributed or are defined by other parts. For example, the fuel tank of an Airbus emerges as the space between the parts of the wing. Similarly, clearances between separate parts (in this case functional subsystems) may be inadequately tracked by the BOM. Some companies advocate a single tree structure BOM, and suffer the consequences of severing the links between parts. Others have multiple BOMs and struggle with the translation between them. Often, individuals, computer programs or formal processes are blamed for problems that most fundamentally arise from trying to map a complex lattice structure to a tree.



7.5 Rolls-Royce Trent series jet engine
Reproduced with the kind permission of Rolls-Royce plc

The structure of relations among parts in a design takes many forms. These structures are dynamic, changing through the process of design.

Mismatches

The structure of relations between parts in a design thus takes many forms. These structures are also dynamic, changing through the process of design as details are specified and performance analysed. However, during the design process it is not only structures of relations between parts which change, but also performance and behaviour of successive design proposals at the various stages of the process.

Analysis at each stage in product development assesses performance or potential performance against specification. Mismatch can occur either in detail or type of behaviour. The former includes mismatch in performance parameters, e.g. fuel consumption or torque characteristics of an engine, whilst the latter includes unexpected behaviour, e.g. vibration resonance from new combinations of design features.

Mismatches in details are handled interactively, whilst mismatches in type resulting from new behaviours emerging in the product during the design process are more difficult to control. Exceptionally, these new behaviours may be desirable – the delightful serendipity of design – but, for the most part, engineering designers try to eliminate these unwanted characteristics.

The later stages of many complex design processes are dedicated to eradicating unwanted behaviour, such as vibration, noise, electrostatic interference (ESI), rumble, heat, *etc*. The design process converges in both these ways to a final design in which behaviour (within the context of use) is predictable and desirable. Surprising and emergent behaviours are evidence of complexity.

An effective process seeks to uncover these behaviours by analysis and test, removing them if possible or restricting the possibility of occurrence by limiting the conditions under which the product is used. In this sense the process seeks to lower complexity of design, especially in the relation between product and user.

Surprising and emergent behaviours are evidence of complexity.

Emergence

Processes for designs like the helicopter rotor blade are also complex because of the structure of many iterative cycles, each with inherent uncertainty, whilst together apparently convergent. The design process may have discernable overall emergent characteristics (such as convergence to satisfactory design) which may not be entirely predictable from the characteristics of its elements. Similarly, designs with internal structural complexity are often intended to behave robustly in a wide variety of contexts. For example, the helicopter rotor blade operates in a specified temperature range and a wide range of altitudes.

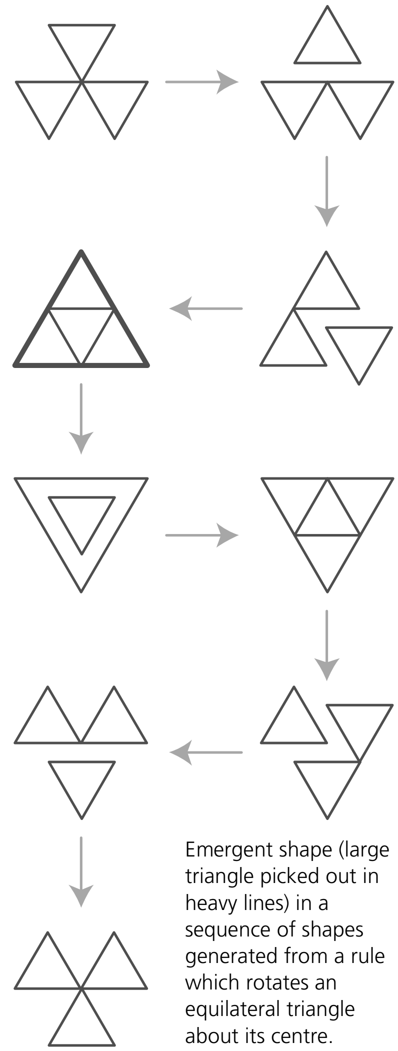
Simon (1969), in his *Sciences of the Artificial*, suggested that an essential aspect of designing takes place at the interface between design and context. However, this interface can be tricky because emergent types of behaviour – that is, surprising and unpredicted behaviour – may arise from interactions of elements or propagation of effects from one area of a design to another.

In the engineering context the aim is usually to reduce this complexity, restricting emergent behaviour of a design to intended function (and of corresponding process to intended outcomes). Designers generally try to avoid emergent behaviour that is random and chaotic by locating designs within margins – for example, compressor stall in a jet engine is avoided using design margins to keep pressure surge within limits.

A major source of complexity arises in the interaction of design and process. Recall the functional and modular groupings in a jet engine considered above. The compressor and turbine are commonly designed by separate teams and there are institutional company barriers to the flow of information, especially change information. Reaction blading changes in the turbine alter axial loads along the rotor, including requirements for compressor bearings and seals. The combination of the effects of design decisions made rationally by individual domain experts may only emerge at prototype test.

On the one hand, decoupling of processes for jet engine design has reduced complexity in designing but increased the complexity in the product and its behaviour, introducing unexpected 'emergent' behaviour. In this case the emergence may be failure of bearings or seals. This example emphasises again the importance of complexity in the relations between major elements in design – in this case product and process.

Although product and process elements are complex in their own right, with many subelements and relations, the major complexity arises from the way that the product lies across the process or, in the language of complexity, forms a 'traffic' through the network of activities and tasks in the design process (Johnson, 1995). As the 'traffic' of product moves through processes the 'product' changes (or strictly speaking its description changes) and new behaviour emerges. So, the three-way relation on the lower part of Figure 7.1 among 'product–user–process' is significant in the complexity of design.



Emergent shape (large triangle picked out in heavy lines) in a sequence of shapes generated from a rule which rotates an equilateral triangle about its centre.

7.6 Exploiting emergence radically changes the shape development (Stiny, 2004)

Uncertainty

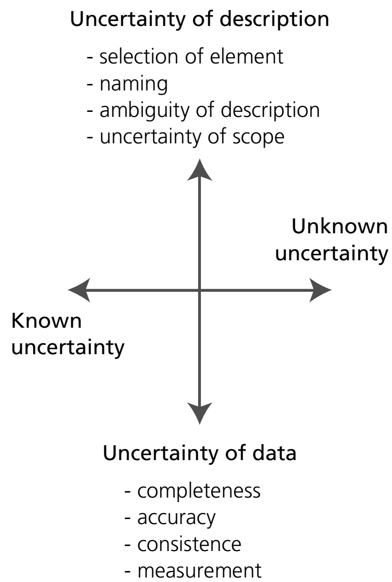
The discussion of examples of complexity in design reveals that complexity arises at several levels in the relations within and among products, processes, users and organisations. We turn to the idea that complexity arises from the balance of uncertainty and order which was introduced in the overview of complexity theory (Figure 7.3 – information measures of complexity).

This information complexity corresponds to entropy, which in its information sense (Jaynes 1957) is a measure of uncertainty relative to constraints (order). Maximising this entropy describes what balance can be expected between uncertainty and order. Complex physical systems seem to balance order and uncertainty at different levels. They might present patterns in overall behaviour but with extensive uncertainty at the microlevel. Conversely, microlevel order may be balanced by surprising overall uncertainties in aggregate behaviour. We would expect a complex design process, although containing a great number of uncertain events, to yield, overall, a satisfactory design. We would expect a complex product with many parts, possibly with uncertain performance early in the design process, to function and meet specification in ordered and predictable ways. Alternatively, a complex product may balance order and uncertainty differently. Extensive uncertainties in operating conditions may be balanced by an ordered behaviour, such as for example in intelligent systems. This section will describe several types of uncertainty which occur in design processes and the counteracting types of order.

Uncertainty is present in all areas of design and designing (products, processes, users, and organisations).

The balance between uncertainty and order can manifest itself in several ways. For example, at the beginning of a design process designers are uncertain about the details of configuration and parameters, but may have a detailed functional specification. So, despite some uncertainty, the specification implicitly restricts the selection of configuration and parameters. Company standards and policies will also direct design choices, thus imposing order, as do past designs and the experiences gathered through them. Without these features of order there would not be complexity. Designers can run out of ideas if there are no constraints. When the product provides few constraints, designers derive them from the wider context. For example, textile designers with few constraints will invoke contexts in prevailing fashion.

Uncertainty is present in all areas of design and designing (products, processes, users, and organisations). New designs have parameters and behaviours which are not known completely beforehand, processes have uncertain durations and uncertain effects, users and conditions of use can change, organisations change and, more widely, contexts, environments and long-term conditions of use are unpredictable. All these uncertainties make planning design processes harder by increasing the numbers and combinations of possible outcomes. Some have argued that uncertainty is at the core of design complexity (Suh, 1999). We will discriminate two basic types of uncertainty: 'unknown' and 'known' uncertainty. These basic types are present in two areas: (i) descriptions and (ii) data (which includes uncertainty in measurement). Similarly, we discriminate between several types of order: structural order of relations between parts, dynamic order of patterns of behaviour and the order imposed by constraints. Generally, complexity seems to occur when there are high levels of uncertainty combined with high levels of order. We focus here on the types of uncertainty (Figure 7.7).



7.7 Types uncertainty in of design

'Known' uncertainty is based on variability in past cases. It can be characterised by probability distributions, e.g. of process task durations or the probabilities of a process (such as a computational analysis or prototype test) improving design performance. A key problem in design is the estimation of these known uncertainties in unique products and processes. Known uncertainties put limits on possibilities and describe them through probability distributions. In other cases, uncertainties may be known but their effects are unknown uncertainties in behaviour.

The uncertainty of surprise is an 'unknown' uncertainty in the sense that there is no particular expectation of such an event. Internal unknown uncertainties arise in the product, the process, the user or the organisation itself. These could range from unexpected material fatigue, to problems with software packages or employees getting pregnant. External unknown uncertainties come from the context in which the product or process operates, such as political events. 9/11 is an extreme example of unknown external uncertainty. Uncertainty in products is one of the sources of uncertainty in process. For example, uncertainty about vibration problems leads to uncertainty in process planning and scheduling. When managers do not know that vibration occurs, they won't plan in resources. When they do not know in which part of the product it will occur, they do not know to which team they need to allocate resources. For example, in helicopter design it is very difficult to predict where and to what extent vibration will occur. It is difficult to know up front what remedial actions will have to be taken and, therefore, what resources will be required. Design managers cope with this by analysing the design as much as possible, but scheduling time at the end of the process for sorting out these, as yet unknown, vibrations.

A key problem in design is the estimation of known uncertainties in unique products and processes.

Both types of uncertainty are present in the uncertainties of descriptions. Designers make choices on those aspects of a design which are to be included in a model and exactly how the model is constructed. For example, the selection of meshing points affects the results of vibration analysis. In modelling, subsystems are grouped together, making analysis within the grouping easier than outside it. The vibration models, for example, look at particular sets of components, but it might be the subtle interaction of these sets of components which causes a problem. These are uncertainties in what is included in the description. During the design process there are also uncertainties in the design itself: in its configuration and parameters and in its behaviour. These are also classified here as uncertainties of description. Besides the uncertainty in the selection of elements there is an ambiguity in how elements are grouped into meaningful concepts. Naming these elements or groups carries its own uncertainties. Each description implies a range of possible meanings, and often the boundaries of the interpretation are uncertain. For example, when a car is called a 'sports car' this may have significantly different meanings for different people. Further, the use of a particular label ('sports car') changes our perception of the design.

Many complex systems are characterised by voluminous hetero-geneous data of variable quality and completeness.

Many complex systems are characterised by voluminous heterogeneous data of variable quality and completeness. Uncertainty in data lies not just in its accuracy but also its completeness and consistency. In design processes and product development, as designs are developed from concept to layout, and then to manufacture, many types of data are generated. Incompleteness is a characteristic of data during design, especially with speculative proposals. In some complex human systems it is impossible to have data that are complete or consistent, and the science of these systems has to accept this as one of its axioms. It is not simply a case of collecting better data to eliminate inconsistency – the issue is to provide robust predictions even though the data are incomplete and inconsistent.

There are underlying 'unknown' uncertainties in all measurements. In chaotic systems, the response to 'unmeasurable' differences in initial conditions is an unknown uncertainty. This randomness is an essential part of how a complex system behaves. But it is not necessarily due to internal uncertainties on the parameters or variables.

Continuous models which are entirely deterministic differential equations can, nevertheless, exhibit wildly random behaviour. In discrete models, state transition probabilities specify the known uncertainties. At a higher level of behaviour, the patterns of these transition probabilities make some types of transition more likely than others. This is the background of known uncertainty against which surprising events and behaviour occur.

Uncertainty is only one of several significant sources of complexity in design. We will now outline some of these in the context of general developments in complexity theory.

7.8 Water. Some systems, such as convection in water, show uncertain behaviour in detail. However, there is emergent, structured and bounded behaviour of the overall system (Nicolis and Prigogine, 1989).

Complexity theory and design

In this section we examine some specific characteristics of complexity which are pertinent to design. While these factors can be managed, they cannot be eliminated, because they are inherent in any complex system.

Dynamics

The main ingredients of deterministic chaos (see Figure 7.3 overview of complexity theory) are (i) sensitivity to initial conditions and (ii) boundedness. The first means that the slightest errors in measuring the initial conditions cause the behaviour to 'explode', but stay within bounds of 'normal' behaviour. Examples include many human and socio-technical systems. Designing and its processes are an example of such hard-to-predict systems. And many products themselves display these characteristics of uncertain behaviour, especially in the context of the wide spectrum of 'users' from the immediate customer to those affected during the design lifecycle and beyond.

Processes of engineering design cope, in practice, with cumulative small effects by redescribing the system at the different stages of the process. Through gateway processes companies force products and processes to reach certain well-defined points. This is a cyclic process of description and prediction. Suh (1999) advocates using this as a design principle for time-dependent systems, such as design processes and schedules. He advocates attempting to transform time-dependent combinatorial complexity (with increasing uncertainties into the future and their 'knock-on' effects) into periodic complexity (with uncertainties being reset at regular intervals). This is achieved by introducing 'gateways' or reducing the dependencies between parts of the design process.

7.9 Clouds. Models of deterministic chaos were initially developed by Lorenz (1963) to model weather patterns.

Understanding the dynamics of many complex systems requires an appropriate notion of time. There is an interplay between the 'calendar' or 'clock' time of physics, and 'system time' defined by the structural 'events' of the system. For example, a product may be planned to be launched on a given day in a given year (calendar time), but the emergent system event "the

7.10 Magic roundabout. Disentangling the paths of connections can improve overall performance, even though the whole system appears more complicated (Johnson, 1976).

product is ready" may not have happened. Such mismatches between system time and calendar time are well known, especially in the software industry. Understanding the complex interplay between events and time is fundamental in design, planning and management.

Connectivity in dynamic systems

A significant aspect of complex system dynamics is the transmission of energy, information and matter, e.g. vehicles and people in transportation systems, information in design teams and goods across the supply chain. These flows require appropriate channels connecting parts of the system. There is a conflict between facilitating essential communication and de-coupling parts of the system to eliminate undesirable interference and noise (as for example in reducing the options offered on a car). Designing an infrastructure 'back-cloth' to carry the system 'traffic' is an essential part of applied complexity theory in planning and management (Johnson, 1995).

Flows take place on networks of connectivities. In design, several types of network may be present:

- product components are connected by function, geometry, manufacture and assembly;
- people, such as engineers, analysts and designers, are connected in team structures, hierarchies and even friendship;
- activities and tasks in the design process are connected by information and design representations, with process interfaces which may operate with checks or as gateways;
- a range of products in a company are connected by shared components, methods of manufacture, designers or design capabilities;
- supply networks include designing, manufacturing and service outsourcing.

The key factor in all these is the flow on the background structure of the connections. Complexity arises from the structure and connectivities of the network, but most importantly from the dynamics of the patterns of flows. However, dynamics can also manifest itself in another way. Some networks change rapidly over the course of a design project. One of these is the network of connectivities among the relevant knowledge of the participants. As the project proceeds, the connectivities will change as knowledge is acquired, analysed and embodied in a design. Other networks, such as the structure of teams, change more slowly during a project. Although connectivities may be present they may not be continuously active but rather are activated by events such as a competitor's new product or a scheduled project meeting.

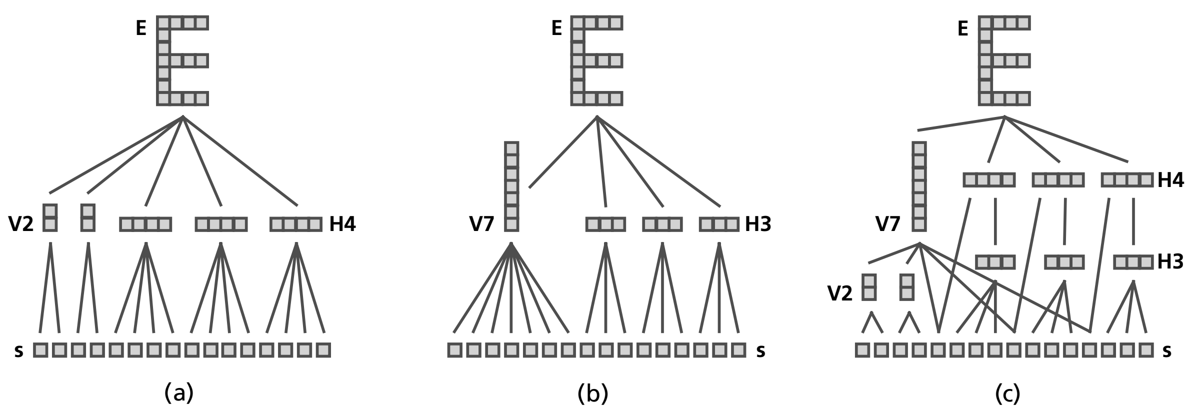
One of the main challenges of design management is to keep an overview of these multiple connections through which information needs to flow, change and propagate (Jarratt *et al*., 2004). A designed backcloth of connectivity, rather than an evolved one, would make it easier to analyse connectivity and the consequences of design decisions. However, this may not be possible, especially where structures are continually reinvented.

One of the main challenges of design management is to keep an overview of these multiple connections through which information needs to flow, change and propagate.
(Jarratt *et al*., 2004)

Structure

Systems are described at different levels of aggregation by structures. To represent and reason about systems it is necessary that the corresponding structured objects have names (Figure 7.11). An important part of the design and management of complex systems involves constructing structured vocabularies. The BOM and other product structures discussed earlier illustrated the problems associated with using a tree structure to map a complex system.

A diagrammatic example with tree and lattice structures. It shows an object given the name **E**. It is made up from a set of 16 atomic objects named **s**. In (a) the squares are assembled into two structures named **V2** and **H4**, and these are assembled into **E**. In (b) the squares are assembled into two structures named **V7** and **H3**, and these are assembled into **E**. Note that superimposing the two structures gives a lattice structure (c) with the squares aggregating in different ways at the middle level. This illustrates that, in general, the intermediate structures (in this case (**V2**, **H4**), and (**V7**, **H3**)) and the names they are given are not unique. There are combinatorially many ways that hierarchies of named components can be constructed to represent a particular object. The selection of a particular hierarchical vocabulary rests with the designer, subject to constraints of how useful it is, and compatibility with pre-existing vocabulary, custom, and culture. It is not uncommon for the vocabulary to have inconsistencies, with the same object having more than one name, or more than one object having the same name. No matter how simple or complex a design, anomalies in vocabulary will increase the complexity, and act as a barrier to effective communication.



7.11 Hierarchies of assembly between a designed object and its atomic components

Hierarchical structures often have many intersections at all levels, leading to a more connected structure called a 'lattice'. Structural descriptions of parts and assemblies in products, or people in teams, fall into such a lattice structure, allowing many different possible groupings. Describing groupings in tree or lattice structures can be problematic in two ways:

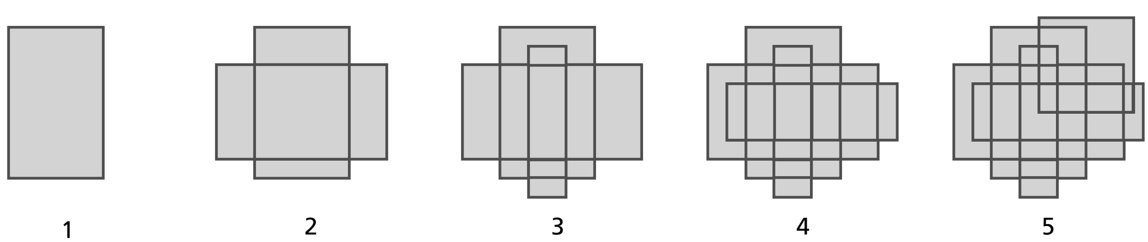
- There are many possible descriptions leading to ambiguity, in the sense that each grouping describes the same object (Stiny, 2004).
- The group elements require names and labels whose construction involves a degree of negotiation. In mature products, naming of parts and groupings is often given by past products. The specific referents in past products can bias the design process significantly.

Search

Many complex systems have large numbers of interacting heterogeneous elements. Looking back at the example of the BOM, there are an enormous number of theoretically possible groupings of the elements in subsystems, which could affect the perception that people have of the BOM.

Figure 7.12 is an example of combinatorial explosion in which adding just one more element under relatively simple relational constraints can generate an order of magnitude more parts and relationships. Much of this emergent structure is not explicitly represented. Many complex systems have large numbers (hundreds, thousands, millions) of heterogeneous elements interacting. These systems do not have simple macroscopic properties, and system behaviour will be driven by micro-agent interactions. For example, an organisation may have hundreds of employees with distinct capabilities from whom to pick teams for projects. There are billions of ways to select six

Combinatorial explosion is illustrated by the graphical example below, where shapes are generated by adding a new rectangle at each stage. Even three rectangles generate dozens more, so that counting them all is a demanding task. It also generates new shapes, such as the inverted U at the top. Adding a fourth rectangle generates even more structure, while by the time a fifth is added, the resultant figure has hundreds of emergent shapes with hundreds of relations between them.



7.12 Small numbers of rectangles generate complex objects with many parts and relations

people from a hundred. Computational search of very large spaces has become an important tool in design and has highlighted the importance of the way problems are represented.

Well-formulated problems have a space of candidate solutions within the representation – the search space – with a subset that are actually solutions – solution space. This simple idea leads to techniques for problem solving based on searching for solutions. When a search space is small, examining every candidate can be a good approach. However, most search spaces are large and may have structures of connections like the lattices above. Exhaustive search is not feasible, so heuristics or random search techniques, such as simulated annealing or evolutionary algorithms, are applied.

The idea of searching for any solution soon leads to the idea of searching for the best or the optimum solution in the search space. Generally, it is impossible or impractical to be sure of finding the best solution to a problem, and optimisation becomes a process in which one seeks relatively good solutions. Design solutions often have to satisfy multiple criteria, so that a robust solution satisfies multiple goals as well as possible. Search is used both to find 'optimal' or satisfactory designs and then to search the possible modes of behaviour for each candidate by varying patterns of inputs and disturbance in a simulation of behaviour. Simulations are an important tool in managing complexity.

Managing complexity

Complexity is often inherent in systems and cannot be eradicated. However, it is possible to take active steps to reduce complexity in the hope of reducing the risk of problems occurring in the design process.

Simulation

The chaotic dynamics of many systems mean that it is impossible to make a point prediction that a certain event will occur at a certain time. Although the behaviour of most complex systems cannot be predicted in detail, there are many things that can be predicted. One answer is the generation of distributions of possible system states emerging from local dynamic interactions (Figure 7.13). Thus, simulations do not give 'point predictions' saying precisely what will happen when, but they give understanding of the spaces of 'possible worlds' in which things may happen, and they give information as to which of all the possible worlds are the most likely to be experienced.

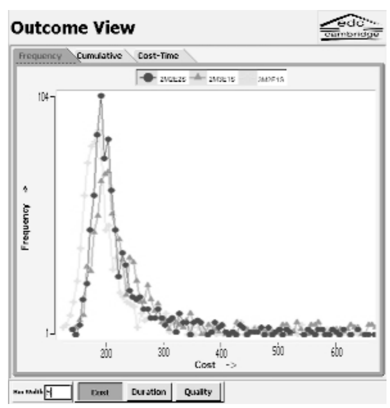


Figure 7.13 Distributions can be obtained by running a simulation many times

As an example of this, consider simulating the behaviour of traffic on a crowded road, including calculating the number of vehicles passing a given point. The actual behaviour of the traffic depends on many things, including the kind of vehicles and the kinds of driver. For these reasons, road traffic dynamics are chaotic, and it is impossible to predict precisely what will happen. Sometimes there will be shock waves as a nervous driver touches the brakes, and sometimes there won't. By simulating the system many times a distribution can be obtained. Although it is impossible to give a point prediction of the flow, such distributions give useful information for the designers of road systems. However, it should be noted that the extent to which simulation results can be trusted (Johnson, 2001) remains a critical issue.

While designers and design managers are well aware of uncertainties in design processes, these are not necessarily accommodated in design-process planning tools.

While designers and design managers are well aware of uncertainties in design processes, these are not necessarily accommodated in design-process planning tools (see Chapter 1 on process models). In reality, the duration and success of tasks are probabilistic. For example, the duration of a particular task or activity in a design process may not be known accurately in advance.

A computational fluid flow analysis may take anywhere between 10 and 100 hours. However, if a similar analysis has been done several times before, then we can consult the distribution of duration times and estimate an expected time for the new task. In design this is always problematic, since the historical data from which the distribution is constructed has not usually been acquired in a controlled way. The activity can change from occurrence to occurrence.

Lessons can be transferred between tasks so that great similarity to previous tasks will reduce both total time and the variation or spread of times expected for a new task. The 'observed' distributions for activity durations can be used in simulations of the whole process or important parts of it, perhaps a set of design tasks undertaken by a smaller team. In turn, a simulation then allows distributions for these sets of tasks to be created.

Simulation is a major tool in design and design process evaluation.

The modes of simulation for complex systems modelling are changing radically. Simulations of large socio-technical systems in areas such as transportation or sustainable development generate models starting from partial and incomplete data and progressively build models guided by convergence (and divergence) between model and practice (Barrett *et al.*, 2001). Simulation is a major tool in design evaluation, and there is considerable potential in using simulation for modelling design processes, and particularly the interactions between designs and their processes (Earl *et al.*, 2001; O'Donovan *et al.*, 2003).

Managing information complexity

Information about synthetic or designed systems is provided by descriptions and representations. One measure of the complexity of existing systems is how extensive their descriptions need to be to capture the features of the design or its behaviour. Algorithmic information theory (Chaitin, 1987) provides the basis for comparing such descriptions. The idea is that designs with compact descriptions, in terms of shorter procedures or fewer rules to generate them, have lower complexity. Designs exhibiting order and regularity in their behaviour may have short descriptions, whilst uncertain and unpredictable behaviour may require longer descriptions. However, taking this to an extreme, if behaviour is random then descriptions again become short as there is little information in the description. An intermediate representation or design proposal, created during the design process, also has an information complexity, although there are additional uncertainties in the design and its parameters. Provided there are statistically reliable estimations of uncertainties or variability, information measures of complexity can identify areas of a design where complexity might be reduced. Information complexity describes the balance between system order and behaviour uncertainty (Figure 7.14)

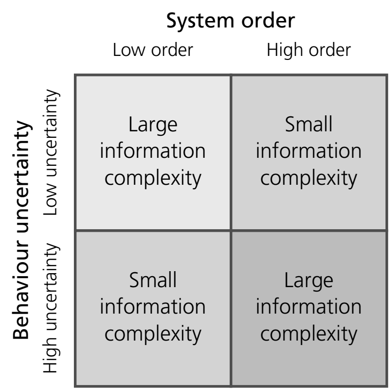


Figure 7.14 System order and behaviour uncertainty

Applying information complexity to the design process is problematic unless uncertainties can be estimated reliably. Many tasks within the process depend on the particular product being designed, the resources available and the 'memory' of similar products. Suh (1999, 2001) takes the view that complexity in design is mainly about uncertainty in parametric assignments. This approach may appear at odds with the idea of information complexity as 'balancing' order and uncertainty. However, with the order of the design process given by functional specification of final design it is feasible to measure complexity of design by uncertainties. In this view, complexity will change continually throughout design as uncertainties change for defined parameters and new parameters are defined and included in the design description.

Examples of complexity and design

The previous sections discussed aspects of complexity that are relevant to different areas of design. Here we describe briefly examples from design which exhibit some of these aspects.

Intelligent systems and control

The compressors of jet aircraft engines use combinations of static and rotating blades to drive air into the combustion chamber. As the blades attached to the rotor pass the fixed stator blades, there is a pressure gain. As with an aircraft wing, the pressure depends on the shape of the blades and their angles. By changing the geometry of the blade the pressure can be increased, but eventually the compressor becomes unstable, with small changes in the control variables causing large and sometimes undesirable changes in pressure. Engineers currently design engines to run 'on the edge of chaos', pushing the parameters to increase performance while (safely) keeping the system out of the dangerous chaotic region. Traditionally the blades were fixed, but some engines have mechanisms to set the angle of the blades optimally for take-off as well as cruising. Currently, a consortium of engineers is investigating the possibility of designing the blades to self-organise, with each blade acting as an agent, selecting its own optimal settings throughout operation of the engine (Johnson *et al*., 2002).

Machines are becoming more intelligent, in terms of being able to sense their environment and respond to it. Cars have navigation systems that know their positions and can compute routes; aeroplanes fly on autopilot. We can expect new types of behaviour as machines communicate with each other, and with remote sources of information (Johnson and Iravani, 2004).

New types of system design have teams or swarms of intelligent machines working together. This approach to the design of engineering systems has many advantages. The members of swarms may self-organise to reconfigure themselves autonomously in order to perform new functions. In almost every area, from toys to domestic products, from industrial machines to transportation systems, designers are building in more autonomous intelligent behaviour.

Manufacturing as a complex system

Uncertainties and variability in manufacturing processes can, to some extent, be controlled.

It is instructive to look at how complexity is modelled in manufacturing systems. Descriptions of processes, relations between processes and dynamic flows through the structure of processes all contribute to understanding the behaviour of manufacturing as a complex system. Uncertainties and variability in manufacturing processes can to some extent be controlled – indeed, the focus on quality in manufacturing processes is about controlling variability in order to deliver a high-quality product to customers at low cost (in the broad sense of resources) to designer and manufacturer. With the possibility of measuring features of manufacturing system behaviour quantitatively (in terms of flows, lead times, inventories and queue size) information-theoretic complexity can be assessed.

Highly predictable processes will have low complexity, as do very variable processes. For these we know little about the overall process – or rather descriptions of what we know are limited. Either the patterns of behaviour are limited or they are so variable that no overall order or regularity can be discerned. We might say that these systems are unlikely to display emergent patterns of behaviour. However, the interesting cases from a complexity point of view are those with a balance of variability and order. Emergent behaviours will occur but the manufacturing system designer will want to limit these to desirable ones.

Static complexity is the "expected amount of information necessary to describe the state of the system" whilst dynamic complexity also includes the "expected amount of information required to report whether a facility is under control".
(Efstathiou *et al.*, 1999)

Complexity reduction by control of processes can increase the effectiveness of manufacturing. The excellent literature on manufacturing system complexity (Deshmukh *et al.*, 1998; Efstathiou *et al.*, 1999; Frizelle and Suhov, 2001) using information-theoretic models is a valuable resource for examining design processes. These models are based on entropy models which measure overall order in systems with high levels of local variability. However, the nature of the local variability is 'known' because processes are repeated and probability distributions can be constructed. However, we note that design is a rather different process, as the local variability is hard to quantify, processes can change and are susceptible to a wide range of external disturbances from customer, suppliers and, last but not least, competing design projects.

Finally, the distinction between static and dynamic complexity may be useful in design. Static complexity is the "expected amount of information necessary to describe the state of the system" whilst dynamic complexity also includes the "expected amount of information required to report whether a facility is under control" (Efstathiou *et al.*, 1999). Although these are useful concepts in understanding complexity, as we have already noted, quantitative information on design process would be required to apply these methods. This prompts a question as to whether this information can be acquired for design or whether design processes are inherently different.

Aerospace engineering design

Aerospace engineering provides illustrative examples of different types of complexity. For complexity arising from the interaction of design and process, the functional and modular groupings in a jet engine have already been considered.

Unexpected interactions between separately designed parts or between new parts and reused parts can also lead to unacceptable overall behaviour. In these cases, although it is in theory possible to analyse the whole design, this is often not done until test prototype. Because of complex multilevel structure and transmission through chains of connection, complexity effects are not picked up until the latter stages of design. Undesirable emergent behaviour is then, if possible, removed.

It is interesting to observe that emergent behaviour arises continuously throughout the process of taking a design from concept to embodiment and manufacture. In some cases this emergence represents new discovery and inspiration for design innovations (as in 'artistic' domains), whilst in other domains, such as engineering, the process of design is to remove undesirable emergent behaviours iteratively. The final design has behaviour which has 'minimal' complexity. This fits nicely with the information view of complexity, since a description of the possible behaviours of a 'well-behaved' design is relatively simple.

Complexity may exist in products, processes, users and management or organisation.

As an example, recall the design of the helicopter rotor blade discussed earlier. The process of design attempts to reduce complexity in behaviour so that it remains predictable. However, at the same time the search for 'optimal' or high-performing designs can lead to parameter values which are in the margins close to where behaviour becomes very unpredictable or chaotic.

Several complexity problems occur here. First, unexpected interactions between parts may cause behaviour to pass over the edges of the margin. Second, it may be that reductions in design process complexity through modularity give this higher design complexity in behaviour. Third, a design has a parameter envelope in which the design performs predictably, but optimal performance often occurs in the margins of this envelope.

Operating in the margins means that behaviour is complex and users require assistance to reduce complexity. An historical example is the comparison of the turning performance of Spitfire and Messerschmitt Me109 aircraft. Theoretically, the Spitfire had better performance in a wider envelope, but Me109s could be flown in narrower margins of their narrower envelope because they incorporated a passive moving element in the wings' leading edges. Although giving only a small aerodynamic improvement, these elements signalled to the pilot that the margin was being encountered. Inexperienced pilots could, therefore, avoid unstable behaviour, reducing complexity and improving performance (Morgan and Morris, 1940).

Conclusions

In this chapter we have shown that design can possess complexity in (a) products, (b) processes, (c) users and (d) designers (their organisation and capabilities). Although each of these elements can be complex, it is their combination that can cause the high levels of complexity that make the design process hard to understand and control.

To design successfully requires that this complexity be recognised and understood. Understanding complex behaviour allows designers and design managers to identify complexity as a root cause of some of their problems and take steps to reduce or manage it. This complexity can be conceptualised and described through a number of formal approaches that give insight into the behaviour of designs and design processes. However, there is no unified theory of complexity and no single theory captures all aspects of a complex system. Despite this limitation, we have shown that light can be usefully shed from differing angles on the problems of design complexity.

References

Barrett C *et al.* (2001) TRANSIMS: transportation analysis simulation system. Technical Report, LA-UR-00-1725, Los Alamos National Laboratory

Batty M, Longley P (1994) Fractal cities: a geometry of form and function. Academic Press

Chaitin GJ (1987) Algorithmic information theory. Wiley

Clarkson PJ, Hamilton JR (2000) Signposting: a parameter-driven task-based model of the design process. Research in Engineering Design, 12(1): 18–38

Cross N (2000) Engineering design methods: strategies for product design. Wiley

Deshmukh AV, Talavage JJ, Barash MM (1998) Complexity in manufacturing systems: part 1 – analysis of static complexity. IIE Transactions, 30(7): 645–655

Earl CF, Eckert CM, Johnson JH (2001) Complexity of planning in design. ASME DETC'01, Pittsburgh, PA, USA

Eckert CM, Blackwell A, Bucciarelli L, Clarkson PJ, Earl C, Knight T, MacMillan S, Stacey M, Whitney D (2004) What designers think we need to know about their processes: early results from a comparative study. Design 2004, Dubrovnik, Croatia

Efstathiou HJ, Tassano F, Sivadasan S, Shirazi R, Alves J, Frizelle G, Calinescu A (1999) Information complexity as a driver of emergent phenomena in the business community. IWES'99, Kobe, Japan

Eppinger SE, Whitney DE, Smith RP, Gebala DA (1994) A model-based method for organizing tasks in product development. Research in Engineering Design, 6: 1–13

Frizelle and Suhov (2001) An entropic measurement of queuing behaviour in a class of manufacturing operations. Proceedings of the Royal Society Series A, 457: 1579–1601

Jarratt TAW, Eckert CM, Clarkson PJ, Stacey MK (2004) Providing an overview during the design of complex products. DCC'04, MIT, Cambridge, MA, USA

Jaynes E (1957) Information theory and statistical mechanics. Physical Review, 106: 620–630

Johnson JH (1976) The q-analysis of road intersections. Journal of Man-Machine Studies, 8: 531–548

Johnson JH (1995) The multidimensional networks of complex systems. In: Networks in action. Springer

Johnson JH (2001) The "Can you trust it? " problem of simulation science in the design of socio-technical systems. Complexity, 6(2): 34–40

Johnson JH, Iravani P (2004) Robotics in the emergence of complexity science. AROB'04, Oita, Japan

Johnson JH, Lucas Smith A, Wiese P (2002) Complexity science for the design of intelligent geometry compressors for jet aircraft engines. AROB'02, Oita, Japan

Lorenz EN (1963) Deterministic non-periodic flow. Journal of the Atmospheric Sciences, 20: 130–141

Mandelbrot BB (1983) Fractal geometry of nature. Freeman

Morgan MB, Morris DE (1940) Messerschmitt Me 109 handling and manoeuvrability tests, Reports and memoranda no. 2361. In: Technical report of the Aeronautical Research Council Special Volume 1, 1955

Nicolis G, Prigogine I (1989) Exploring complexity. Freeman

O'Donovan BD, Clarkson PJ, Eckert CM (2003) The simulated design process. ICED'03, Stockholm, Sweden

Simon H (1969) Sciences of the artificial. MIT Press

Stiny G (in press) Shape: an essay on rules, ambiguity and design. MIT Press

Suh NP (1999) A theory of complexity, periodicity and the design axioms. Research in Engineering Design, 11(2): 116–132
Suh NP (2001) Axiomatic design – advances and applications. Oxford University Press
Wilson A (2000) Complex spatial systems: the modelling foundations of urban and regional analysis. Prentice Hall
Wolfram SA (2002) New kind of science. Wolfram Media