Chapter 3
Theoretical Framework

This chapter introduces a generic approach to carry out protocol analyses of designers using the Function-Behavior-Structure (FBS) ontology. It suggests codifying design protocols into FBS design issues and deriving FBS design processes using two models, a syntactic model and a semantic model. The syntactic model assumes that any design issues is cognitively related to its immediately preceding issue and as a consequence there is a design process; the concept of using Markov analysis is also presented as a tool to examine the syntactic model. Semantic design processes are derived from ontologically coded linkographs. The construction of the linkograph is further examined in this chapter, as it is the foundation upon which further concepts are built. The information captured in the linkograph is studied using statistics and clustering. The rationale of using information theory, entropy, to measure the linkograph is presented, as well as a concise explanation of information theory.

3.1 Design Ontology and Ontological-Based Coding

This section explores the use of the FBS ontology (Gero 1990) to develop a general coding scheme. Its aim is to capture semantic information from design protocols. This semantic information can then be utilised: (1) to explore different aspects of designing according to the focus of interest; (2) to quantify the use cognitive resources; and (3) to locate different types of design transformation processes.

J.W.T. Kan and J.S. Gero, *Quantitative Methods for Studying Design Protocols*, DOI 10.1007/978-94-024-0984-0_3

3.1.1 FBS Ontology and Coding

The FBS framework (Gero 1990) models designing in terms of three classes of ontological variables: function, behaviour and structure. In this view the goal of designing is to transform a set of functions into a set of design descriptions (D). The function (F) of a designed object is defined as its teleology; the behaviour (B) of that object is either derived (Bs) or expected (Be) from the structure, where structure (S) represents the components of an object and their relationships. A design description is never transformed directly from the function but undergoes a series of processes among the FBS variables. These processes include: formulation which transform functions into a set of expected behaviours; synthesis, wherein a structure is proposed to fulfil the expected behaviours; an analysis of the structure produces derived behaviour; an evaluation process acts between the expected behaviour and the behaviour derived from structure; and documentation, which produces the design description. Based on the structure there are three types of reformulation: reformulation of structure, reformulation of expected behaviour and reformulation of function. Reformulation of function is relatively rare, as it changes or redefines the design problem. Figure 3.1 shows the relationships among the eight transformation processes and the three basic classes of variables. The problem space and solution space are expanded by the introduction of new

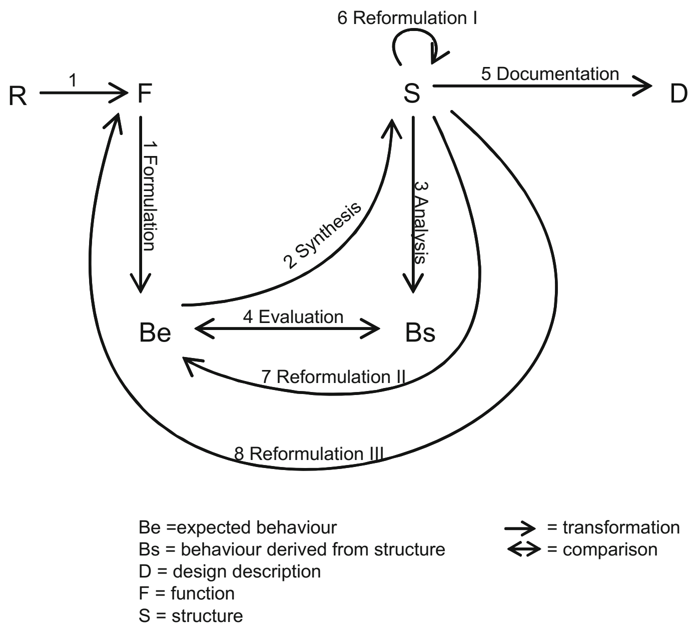


Fig. 3.1 The FBS ontology of designing

variables. These variables are introduced in the reformulation processes; structure, behaviour and function can all be part of reformulations.

The proposed generic coding scheme consists only of the function (F), expected behavior (Be), behavior derived from structure (Bs), structure (S), documentation (D) and requirement (R). Documentation and requirement are both describable in terms of function, behaviour or structure and do not require an extension of the FBS ontology. The protocols are segmented strictly according to these six categories. See Gero and Kannengiesser (2014) for a fuller explanation of the FBS ontology. Part of Gero and McNeill's (1998) coding scheme concerns the designer's reasoning about function, behaviour or structure in the problem domain. They do not separate the expected and derived behaviour.

3.1.2 Situated FBS Ontology

A number of new concepts constitute the situated FBS framework: the notion of situated cognition introduced by Clancey (1997); the idea of constructive memory based on the work of Dewey (1896) and Bartlett (1932); and the observation of designing as an "interaction of making and seeing" by Schön and Wiggins (1992). Gero and Kannengiesser (2002, 2004) developed these ideas further and integrated them into the FBS ontology to form the situated FBS framework by introducing interactions among three worlds—the external, interpreted and expected worlds. A brief description is provided here, however, for a complete exposition readers should consult the original papers (Gero and Kannengiesser 2002, 2004). A designer interacts and understands the external world through her/his interpretation of the external world to form memories of her/his interpreted world in terms of the FBS variables. In order to change the external world (the act of designing) s/he "focuses" to transform experiences to produce the expected world (also in terms of FBS) before taking action in the external world. In this framework the original eight processes are increased to twenty to allow for these additional activities.

Figure 3.2 presents the situated FBS ontology of designing. In the figure, R represents the requirement which is being interpreted in terms of function (F^i), behaviour (B^i), and structure (S^i). In the following text this interpretation process is represented by the symbol "↺". In the interpreted world there are four types of processes that the FBS variables can go through: transformation, represented by "→"; comparison, represented by "↔"; reflection or re-interpretation, represented by "↺"; and focusing, represented by "⇔". Focusing (⇔) refers to processes that produce an expected function (Fe^i) from an interpreted function (F^i), expected behaviour (Be^i) from interpreted behaviour (B^i), and expected structure (Se^i) from interpreted structure (S^i). Expected structure (Se^i) can also be transformed (→) from expected behaviour (Be^i), which in turn can be transformed from expected function (Fe^i), which represents the synthesis and formulation process in the original FBS

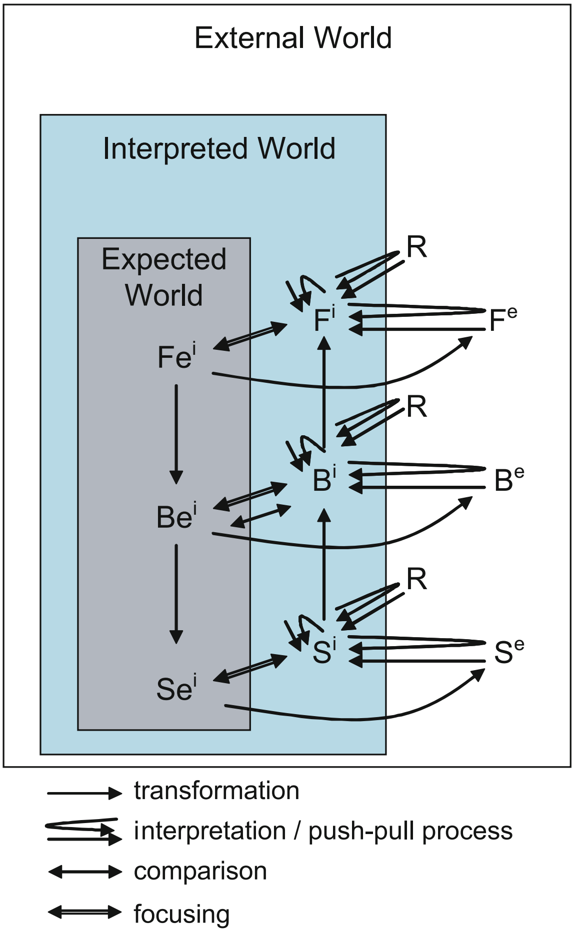


Fig. 3.2 The situated FBS ontology of designing

framework. The comparison (↔) is between expected behaviour (Be^i) and interpreted behaviour (B^i), which is similar to the evaluation in the original FBS framework.

Table 3.1 relates the twenty situated FBS processes to the original eight processes. Of particular interest are the formulation and reformulation processes in this framework. The formulation process involves: the interpretation of requirements (R) in terms of F^i, B^i, and S^i representations ($R \circlearrowleft F^i$, $R \circlearrowleft B^i$, $R \circlearrowleft S^i$); reflecting, based on experience, on those representations ($F^i \circlearrowleft F^i$, $B^i \circlearrowleft B^i$, $S^i \circlearrowleft S^i$, $F^e \circlearrowleft F^i$, $Be \circlearrowleft Bi$, $Se \circlearrowleft Si$); focusing on subsets of these internalised requirements ($F^i \Leftrightarrow Fe^i$, $B^i \Leftrightarrow Be^i$, $S^i \Leftrightarrow Se^i$); and the process $Fe^i \rightarrow Be^i$ that corresponds to the original formulation in the FBS framework. Focusing and reflecting ($F^i \Leftrightarrow Fe^i$, $B^i \Leftrightarrow Be^i$, $S^i \Leftrightarrow Se^i$) appear in all the three types of reformulations. The reformulations II and III are not limited to be driven by structure alone, but also by external representations of function ($F^e \circlearrowleft F^i$) and behaviour ($B^e \circlearrowleft B^i$).

Table 3.1 Situated FBS processes in relation to the eight FBS processes

1. Formulation	R↺F^i, R↺B^i, R↺S^i, F^i↺F^i, B^i↺B^i, S^i↺S^i, F^i ⇔ Fe^i, B^i ⇔ Be^i, S^i ⇔ Se^i, Fe^i → Be^i
2. Synthesis	Be^i → Se^i, Se^i → S^e
3. Analysis	S^i → B^i
4. Evaluation	B^i ↔ Be^i
5. Documentation	Se^i → S^e, Be^i → B^e (optional), Fe^i → F^e (optional)
6. Reformulation I	S^i ⇔ Se^i, S^i↺S^i, S^e↺S^i
7. Reformulation II	B^i ⇔ Be^i, S^i → B^i, B^e↺B^i, B^i↺B^i
8. Reformulation III	F^i ⇔ Fe^i, B^i → F^i, F^e↺F^i, F^i↺F^i

→ = transformation
↔ = comparsion
⇔ = focusing
↺ = interpretation, push-pull process

In protocol studies, designers can only be observed from the external world, the interpreted and expected world are all internal to the designers and can only be inferred from their protocols. Their actions are interpreted by the coder, so there is a degree of subjectivity in the analysis.

An example will be given later in this chapter to elaborate the situated FBS and FBS coding scheme. Before that, in the next section, an application of the notion of a general coding scheme for designing will be presented to show its potential contribution towards design research.

3.2 Meta-Analysis of Design Protocols Based on FBS Ontological Coding

As the FBS ontological coding scheme is a general coding scheme, it is possible to do meta-analysis of design protocols across domains. In order to investigate the commonalities across design domains, Gero et al. (2014a) proposed using the cumulative occurrence of design issues as a basis to examine the relative cognitive design effort across a design session. Cognitive design effort refers to the cognitive activities associated with designing. We will shorten the term to cognitive effort in this book. The cumulative occurrence of design issues models the cumulative cognitive effort across that design session. The cumulative occurrence of a design issue across all segments in a design protocol is calculated as follows: the cumulative occurrence (c) of design issue (x) at segment (n) is $c = \sum_{i=1}^{n} x_i$ where (x_i) equals 1 if segment (i) is coded as (x) and 0 if segment (i) is not coded as (x). Plotting the results of this equation on a graph with the segments (n) on the

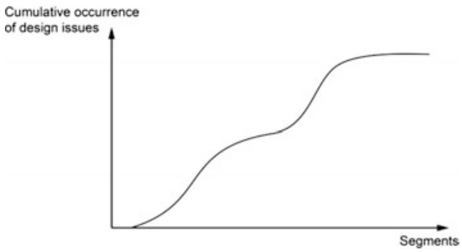


Fig. 3.3 Exemplary graph representing the cumulative occurrence of design issues (original caption of Gero et al. 2014b)

horizontal axis and the cumulative occurrence (c) on the vertical axis yields a visual representation of the cumulative cognitive effort represented by the occurrence of the design issues in a protocol, Fig. 3.3 (Gero et al. 2014a).

Based on the notion of cumulative occurrence of design issues, Gero et al. (2014a) utilised the following qualitative measures for each of the six classes of design issues:

- First occurrence at start: Which design issues first occur near the start of designing, and which first occur later?
- Continuity: Which design issues occur throughout designing, and which occur only up to a certain point?
- Shape of the graph: For which design issues is the cumulative occurrence graph linear, and for which is it non-linear?
- Slope: This is a measure for the speed at which design issues are generated.
- R^2 (coefficient of determination): This is a measure for the linearity of the graph. We will set a minimum value of 0.950 as a condition for linearity.

All of the above measures are independent of the length of the design session, which allow comparison of design protocols with different numbers of segments.

Gero et al. (2014a) used this model to examine 13 sets of design protocols drawn from a variety of studies carried out by different researchers in different countries involving design task and different levels of expertise. They found some commonalities, which is not surprising, given existing assumptions, observations and hypotheses about designing. For example, the design process commences with clarifying a set of requirements and functions that was shown by the discontinuous graphs of these two issues. However, some of the findings are unexpected and might bring new insight to the theory of designing. For example the structure issues occur continuously throughout design sessions and occur at a linear rate, Figs. 3.4 and 3.5 show the cumulative occurrence of structure issues of mechanical engineering students and a professional design team respectively. The results showed this linear cumulative occurrence of structure issues in all the 13 sets of design protocols. This is contrary to the notion that a design process start with requirements and functions and then, at a later stage, ends with structures and descriptions.

Gero et al. (2014b), using the same notion of cumulative occurrence of issues, examine three sets of data; the first set of data contains 18 design protocols of mechanical engineering students at different stages in design education; the second

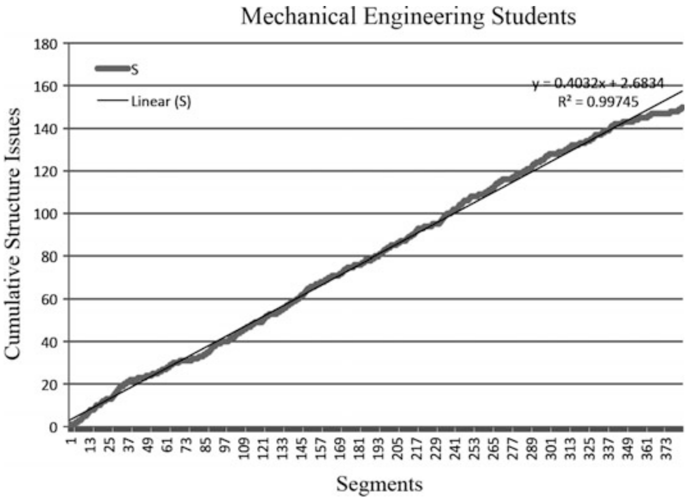


Fig. 3.4 Cumulative occurrence of structure issues of mechanical engineering students (after Gero et al. 2014a)

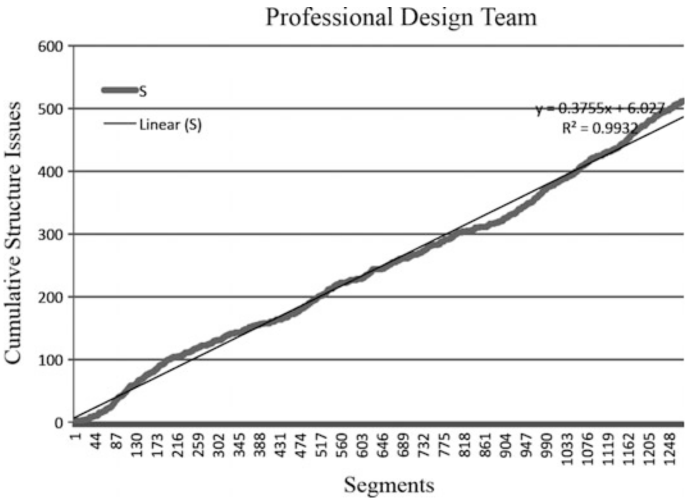


Fig. 3.5 Cumulative occurrence of structure issues of a professional design team (after Gero et al. 2014a)

set of data contains 31 design protocols of mechanical engineering students being taught different concept-generation methods; and the third set of data contains 42 sessions of software engineering students. Their results provide evidentiary support of the linearity of structure issues in all three datasets and in that there is no statistically significant differences between the mean slopes of the linear graphs. This implies that the cognitive effort expended on structure is expended uniformly across all the design sessions independent of the domain and task. Further, the rate of expenditure is independent of domain and task for these three studies of mechanical engineering students.

Kannengiesser et al. (2015), in a longitudinal study, using the same meta-analysis notion tested if design cognition of high school students who have taken pre-engineering courses (experiment group) would be different to those who have not (control group). Again, they found the same linearity of structure issues for the two groups.

The aim of this sub-section is to demonstrate the potential and significance of a general coding scheme so results of the studies described above are not fully described here.

3.3 Linkography

Linkography was first introduced to protocol analysis by Goldschmidt (1990) to assess the design productivity of designers. The design protocol is broken down into small units called "design moves". Goldschmidt defined a design move as "a step, an act, an operation, which transforms the design situation relative to the state in which it was prior to that move" (Goldschmidt 1995, p. 195) or "an act of reasoning that presents a coherent proposition pertaining to an entity that is being designed" (Goldschmidt 1992, p. 72). The following example is taken from Goldschmidt (1992). The four moves below are accompanied by a sketch, illustrated in Fig. 3.6.

1. If I look at the form again (it might also be the influence of having done entry 1, but) it seems that spatially, these are the larger directions (w, x).
2. I am getting one, two, three spaces here (p) and one, two (q) there.

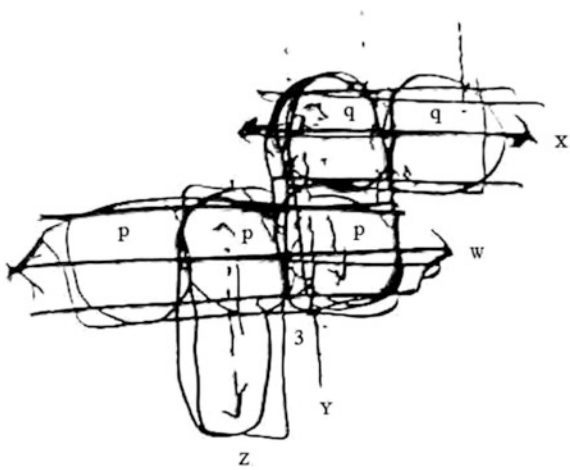


Fig. 3.6 Sketch from a protocol. Goldschmidt (1992), original caption

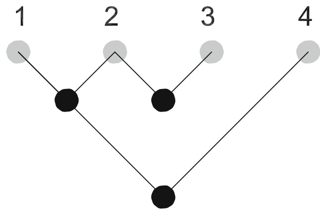


Fig. 3.7 Goldschmidt example of linkograph with four moves with the addition of nodes (shown in *black*) used in subsequent quantitative analyses

3. They're about square, so there is a tendency to try and see them as spaces.
4. These are secondary directions within the space (y, z), so the entry (3) is actually moving in along the secondary directions.

A linkograph is then constructed by linking related moves. The links are established by discerning, using domain knowledge and common sense, whether a move is connected to the previous moves. In her exposition, move 2, if judging from the verbalisation only, is not linked to move 1, but looking at the sketch, "the spaces it specifies are articulated through encircling the large directions of move 1" so move 2 is linked to move 1. Move 3 elaborates on move 2 but does not seem to link with move 1. Move 4 "discontinues the spatial diagnosis of moves 2 and 3" and returns to "the question of directions which was first brought up in move 1". Figure 3.7 is constructed by joining the linked moves. It can be seen as a graphical network of associated moves that represent the design session.

The design process can then be examined in terms of the patterns of move associations. Goldschmidt identified two types of links: backlinks and forelinks. Backlinks are links of moves that connect to previous moves. Forelinks are links of moves that connect to subsequent moves. In Fig. 3.2, moves 2 and 4 are backlinked to move 1 and move 3 is backlinked to move 2; move 1 is forelinked to moves 2 and 4, and move 2 is forelinked to move 3. Conceptually forelinks and backlinks are very different. Goldschmidt (1995, p. 196) stated, "backlinks record the path that led to a move's generation, while forelinks bear evidence to its contribution to the production of further moves."

The link index and critical moves were devised as indicators of design productivity. A link index is the ratio between the number of links and the number of moves; in this case it is 3/4. Critical moves are design moves that are rich in links. Figure 3.8 is another linkograph from Goldschmidt (1995), with six critical moves. They can be forelinks (moves 21, 26, and 28), backlinks (moves 15, 31, and 32), or both (moves 26 and 31). In her exposition, design productivity is positively related to the link index and critical moves; a higher value of link index and critical moves indicates a more productive design process. Later, Goldschmidt and Tatsa (2005) provided empirical evidence that quality outcome and creativity hinge on good ideas or what she called critical moves.

With an understanding of the construction of a linkograph, one is able to comment on the design behaviour without looking at the design protocol. Goldschmidt (1992) suggested that the linkograph pattern of productive designers will be different from that of less productive designers. Productive designers will

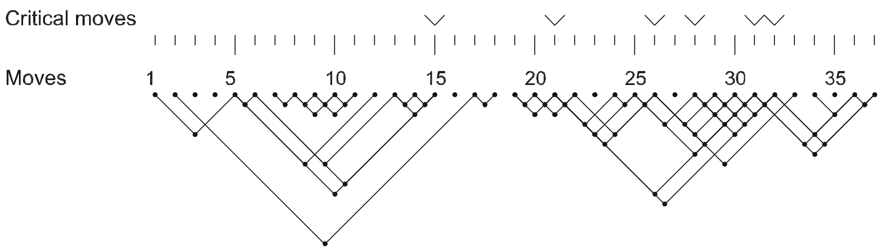


Fig. 3.8 Linkograph from Goldschmidt (1992); "v" indicates critical moves

elicit moves that have a high potential for connectivity to other moves, while less productive designers will exhibit random trails with moves that do not have high potential for contribution to the design concept. In addition, designers who start the design process by exploring different options and then select one to develop will produce a very different linkograph compared to designers who use a holistic approach without exploring different options.

3.4 Syntactic Design Process

We define syntactic design processes as the transformation of cognitively related design issues by assuming that any design issue is related to its immediately preceding issue. As we can see from the above subsections on linkography this assumption is not necessarily the case. In the Sect. 3.5, we define semantic design processes as design processes that are derived by considering the semantic linkage of design issues. We propose using the syntactic design processes as an efficient way to link design issues to produce design processes as it does not involve the labour intensive construction and arbitration of linkographs.

3.4.1 FBS-Based Design Issues of an Episode

If the FBS ontological coding scheme is used to study the issues of the four-move example depicted in Sect. 3.3, move 1 ("If I look at the form again it seems that spatially, these are the larger directions") will be coded as behaviour (Be), since it reasons the spatial behaviour of the form as directions. Move 2 ("I am getting one, two, three spaces here and one, two there") will be structure (S), because it describes spaces and their topology. Move 3 ("They're about square, so there is a tendency to try and see them as spaces") will be behaviour (Bs), as it rationalises and analyses the behaviour of the squares. Move 4 ("These are secondary directions within the space, so the entry is actually moving in along the secondary directions") will be behaviour (Be) again, as this move concerns the directional aspects of the spaces and the entrance. The sequence of the design issues is Be, S, Bs, and Be.

3.4.2 Syntactic Design Process of the Episode

In this book, processes are derived either syntactically or semantically. With the four design issues here we will have three syntactic processes. Be–S, which is a synthesis process, S–Bs, an analysis process and Bs–Be, which is considered an evaluation process.

3.4.3 Situated FBS-Based Design Issues of an Episode

Again, Goldschmidt's linkograph example in Fig. 3.6 is used to expound how the situated FBS framework can be used as a coding scheme. As each of the four moves contains more than one category of design issue, for example, accompanied with drawing actions, they have to be re-segmented. To avoid confusion, the new segments are called segments instead of moves. The 10 coding categories correspond to situated FBS framework are:

1. R: given requirements or derived from the brief
2. F^i: interpreted function either derived from requirements or ascribing meaning to the depicted structure
3. F^e: external representation of function, usually expressed in written words
4. Fe^i: expected function resulting from focusing on the interpreted function
5. B^i: interpreted behaviour from the depicted structure or requirements
6. B^e: external representation of behaviour, in terms of symbols or written words
7. Be^i: expected behaviour derived from the expected function or interpreted behaviour which result(s) from the requirements or interpreted structure
8. S^i: interpreted structure either from the external structure or from requirements
9. S^e: depiction that indicates the structure
10. Se^i: expected structure from expected behaviour or by focusing on the interpreted structure, sometimes without depiction

Assuming the interpretation of the sketches prior to the verbal protocol is segment 1 and is coded as S^i, the original move 1 is triggered by the designer looking at the sketch (form) again (re-interpreting the structure of segment 1) and then drawing the horizontal axis/direction *w*, *x*, which is a documentation of behaviour. This move is broken down into three segments:

2. "If I look at the form again it seems that spatially…"

This segment is a re-interpretation of behaviour of existing drawings, so it is coded as B^i and it is connected to segment 1.

3. "…these are the larger directions…"

This is coded as Be^i, because the designer is expecting the spatial behaviour of the structure in terms of direction. This segment is related to both segments 1 and 2.

4. *(Draw w x lines)*

This action is coded as S^e because it is a depiction of the spatial relationship, the topology of the structure. When considering the connections, this action is linked to the expected behaviour (previous segment 3) and the topology of the previous structure depiction (segment 1). This segment can be refined into two separated actions of depicting behaviour, but in this illustration there is no point in doing so, because *w* and *x* have not been referred to individually.

Move 2 is about reading the structure as five spaces, which can be further refined into five separate actions of depicting the structure (circling the 5 squares). Again, it is not separate and is truncated into two segments, one expectation of structure and one drawing action.

5. "I am getting 1, 2, 3 spaces here and 1, 2 there"

This segment is a re-interpretation of the depicted structure, which is related to the interpreted structure in segment 1 so it is coded as S^i, it is also related to segment 2 of "… look at form again… spatially" and the drawings in segment 1. This segment does not seem to relate to the directional aspect of segments 3 and 4.

6. *(Draw p and q squares)*

This segment is coded as S^e as it is about the form of the building, which is linked to the expected structure in segment 5 and the sketch in segment 1, because the sketch sets the boundary of the "spaces".

Move 3 is about justifying the spaces that the designer has just circled. This move is re-segmented into two segments, one interpretation and one expectation.

7. "They're about square…"

is coded as S^i, as it concerns the form. It is linked to segments 5 and 6, as the designer is focusing on *p* and *q* but not the overall form, so it is not linked to segments 1 and 2, nor does it link to segments 3 and 4 which are about axis and direction.

8. "so there is a tendency to try to see them as space"

is coded as Se^i, as the "… try to see them as space" is an expectation of the form. It is linked to segments 5, 6 and 7, as the state of affairs concerns the *p* and *q* spaces and not other things.

Move 4 returns to the directional aspect of the designed spaces and contains drawing actions. It also reflects on the influence of the axis towards the entrance. It has been divided into three segments.

9. "These are secondary directions within the space…"

This is a re-interpretation and expectation of the spatial behaviour of the design, so it is coded as Be^i. It is connected to segments 1, 2, 3 and 4 because they are all related to the orthogonal axis of direction.

10. *(Draw y z lines)*

As in segment 4, it is coded as S^e. It is a result of the above segment (9) and it hinges on previous depictions, so it is linked to segments 1 and 4.

Table 3.2 Segments coded with situated FBS issues

1	S^i	Structure before move 1
2	B^i	If I look at the form again it seems that spatially
3	Be^i	These are the larger directions
4	S^e	Draw w x
5	Se^i	I am getting 1, 2, 3 spaces here and 1, 2 there
6	S^e	Draw p and q
7	S^i	They're about square
8	Se^i	So there is a tendency to try to see them as space
9	Be^i	These are secondary directions within the space
10	S^e	Draw y z
11	B^i	So the entry (3) is actually moving along the secondary directions

11."…so the entry (3) is actually moving along the secondary directions."

This segment is coded as B^i, as the designer discovers the directional behaviour of the entrance. This seems to be only related to the idea of the secondary (*y*) axis the designer has just raised, so it is linked only to segments 9 and 10.

Table 3.2 shows the re-segmented protocol and their associated coded issues. This protocol does not contain examples of function or requirement coding categories.

3.4.4 Using Markov Chains to Describe the Design Process

This section uses the situated FBS-coded segments to illustrate some concepts of Markov analysis. In this analysis the links in linkographs are not considered; only the sequence of the design issues is used. Markov chains, also referred to as Markov analysis and Markov models, examine the sequence of events; they analyse or describe the probability of one event leading to another. In mathematics, a Markov chain is a discrete-time stochastic process with a number of states such that the next state solely depends on the present state. Markov chains have been used to analyse writers' manuscripts and to generate dummy text (Kenner and O'Rourke 1984); for the ranking of web pages by Google (Langville and Meyer 2006); and to capture music compositions and synthesise scores based on the analyses (Farbood and Schoner 2001).

In protocol analysis, McNeill et al. (1998), treating analysis, synthesis and evaluation as Markov states, found that the most likely event to follow analysis is a synthesis event. Also the most likely event after synthesis is an evaluation event and the most likely event after an evaluation event is a synthesis event.

To illustrate a Markov chain, each design issue of a segment is considered as an event. The purpose is to investigate the sequence of the events (coded segments) in relation to the probability of the previous events. The simplest Markov chain is the

Table 3.3 Percentage of one state to the next state

State	Next state				
	Be^i (%)	B^i (%)	Se^i (%)	S^e (%)	S^i (%)
Be^i				100	
B^i	100				
Se^i	50				50
S^e		33	33		33
S^i		50	50		

first-order chain which only examines the intermediate state after an event. In this example there are 11 segments, so there are 10 transitions from one state to another. Within the 11 segments there are only five different design issues being coded, B^i, Be^i, S^e, Se^i and S^i. Table 3.3 shows the percentages of each event in one state in relation to the next state. With this observation of events it can be turned into a transition matrix, Eq. 3.1, which represents a Markov process. The numbers in the matrix represent the probability of an event. Alternatively, this can be represented in graphic form, as illustrated in Fig. 3.9. This figure demonstrates that when the event B^i happens, the next event will be Be^i ($p(Be^i) = 1$). The event after Se^i will share a 50 % chance of being Be^i or S^i.

$$P = \begin{pmatrix} & Be^i & B^i & Se^i & S^e & S^i \\ Be^i & 0.00 & 0.00 & 0.00 & 1.00 & 0.00 \\ B^i & 1.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ Se^i & 0.50 & 0.00 & 0.00 & 0.00 & 0.50 \\ S^e & 0.00 & 0.33 & 0.33 & 0.00 & 0.33 \\ S^i & 0.00 & 0.50 & 0.50 & 0.00 & 0.00 \end{pmatrix} \quad (3.1)$$

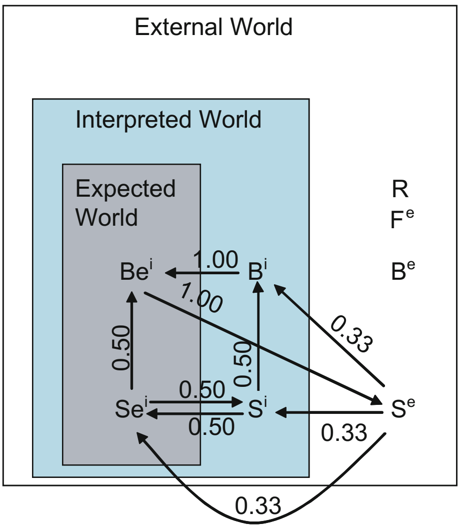


Fig. 3.9 The probability from one FBS state to another FBS state

3.4.5 Some Properties of Markov Chains

Kemeny and Snell (1960) classified three types of Markov chains based on their behaviours: absorbing, regular and ergodic. An absorbing chain is one that consists of states that, once entered, will never be left. This is not likely to happen in design protocols. For example if S is the absorbing state, once the sequence has reached this state, all the following states will be S and nothing else.

A chain is regular if and only if it is possible to be in any state after a number of steps no matter what the starting state is. A chain is ergodic only if it is possible to transit directly from a state to any other state. Once a system enters an ergodic set it will never leave it. Here, only the properties of regular chains will be considered.

Probability Matrix/Vector of Regular Chains

Some behaviours of a regular Markov chain are:

the powers of P^n approach a probability matrix A;
each row of A is the same probability vector $\alpha = a_1, a_2, \ldots, a_n$; and
for any probability vector π, $\pi \cdot P^n$ approaches the vector α as n approaches infinity.

$$\text{i.e. } \alpha P = \alpha \tag{3.2}$$

Essentially these mean that there is a limiting probability a_j of being in the state s_j independent of the starting state. Using the above example and the equation $\alpha P = \alpha$, P can be substituted by Eq. 3.6 to obtain the following five equations.

$$a_2 + \frac{1}{3}a_3 = a_1$$
$$\frac{1}{3}a_4 + \frac{1}{2}a_5 = a_3$$
$$a_1 = a_4$$
$$\frac{1}{2}a_3 + \frac{1}{3}a_4 = a_5$$

Since α is a probability vector the sum of the elements equals one.

$$a_1 + a_2 + a_3 + a_4 + a_5 = 1$$

The unique solution to these equations is:

$$a = \left(\frac{1}{4}\frac{1}{6}\frac{1}{6}\frac{1}{4}\frac{1}{6}\right) \tag{3.3}$$

This means that for a large number of coded segments one can expect $\frac{1}{4}$ (a_1) of the segments will be Be^i, $\frac{1}{6}$ (a_2) of the segments will be B^i, $\frac{1}{6}$ (a_3) of the segments will be Se_i, $\frac{1}{4}$ (a_4) of the segments will be S^e, and $\frac{1}{6}$ (a_5) of the segments will be S^i. This distribution of design issues is a little different from the original distribution

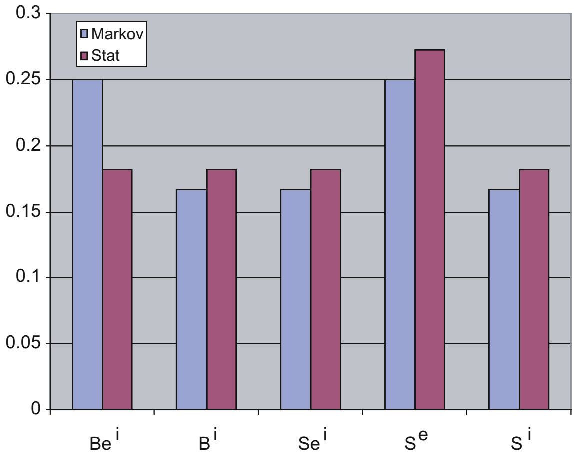


Fig. 3.10 Charts showing the code distribution probabilities based on Markov analysis and statistics

Fig. 3.10 shows the differences. The chart suggests that, when there is a large number of coded segments, the Markov analysis predicts more Be^i design issues than the statistical prediction, whereas the Markov prediction of the occurrence of other segments is lower than the prediction by statistical analysis.

Traditional protocol analysis is based heavily on statistical analysis: this contains the assumption that each segment is an independent event. Markov analysis, based on the probability of relationship with the last event, provides another venue for insight into the design activities.

First Passage Times

The mean first passage time is the average number of steps traversed before reaching a state from other states. The mean passage time can be obtained from the transition matrix and the probability matrix. Kemeny and Snell (1960) proved that the mean first passage matrix M is given by:

$$M = \left(I - Z + EZ_{dg}\right)D \tag{3.4}$$

where I is an identity matrix, E is a matrix with all entries of 1, D is the diagonal matrix with diagonal elements $d_{ii} = 1/a_i$, and Z is the fundamental matrix such that

$$Z = (I - (P - A))^{-1} \tag{3.5}$$

From Eq. 3.8

$$A = \begin{pmatrix} \frac{1}{4} & \frac{1}{6} & \frac{1}{6} & \frac{1}{4} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{6} & \frac{1}{6} & \frac{1}{4} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{6} & \frac{1}{6} & \frac{1}{4} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{6} & \frac{1}{6} & \frac{1}{4} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{6} & \frac{1}{6} & \frac{1}{4} & \frac{1}{6} \end{pmatrix} \tag{3.6}$$

Put Eqs. 3.6 and 3.1 into Eq. 3.5

$$Z = \begin{pmatrix} 0.69 & -0.04 & -0.04 & 0.44 & -0.04 \\ 0.44 & 0.79 & -0.21 & 0.19 & -0.21 \\ 0.10 & -0.10 & 0.90 & -0.15 & 0.24 \\ -0.06 & 0.13 & 0.13 & 0.69 & 0.13 \\ 0.02 & 0.18 & 0.18 & -0.23 & 0.85 \end{pmatrix} \quad (3.7)$$

Solving Eq. 3.4 the following matrix is obtained.

$$M = \begin{pmatrix} & Be^i & B^i & Se^i & S^e & S^i \\ Be^i & 4.00 & 5.00 & 5.67 & 1.00 & 5.33 \\ B^i & 1.00 & 6.00 & 6.67 & 2.00 & 6.33 \\ Se^i & 2.33 & 5.33 & 6.00 & 3.33 & 3.67 \\ S^e & 3.000 & 4.00 & 4.67 & 4.00 & 4.33 \\ S^i & 2.67 & 3.67 & 4.33 & 3.67 & 6.00 \end{pmatrix} \quad (3.8)$$

Thus, for example, if the designer is at the $\mathrm{Be^i}$ state, the mean number of steps before another $\mathrm{Be^i}$ state is 4, the mean number of steps before a $\mathrm{B^i}$ state is 5; before an $\mathrm{Se^i}$ state is 5.67; before an $\mathrm{S^e}$ state is 1; and before an $\mathrm{S^i}$ state is 5.33.

In this example, the shortest paths are $\mathrm{Be^i}$ to $\mathrm{S^e}$ and $\mathrm{B^i}$ to $\mathrm{Be^i}$ (1 mean step) and the longest one is $\mathrm{B^i}$ to $\mathrm{Se^i}$ (6.67 mean steps). Since this example has limited observations, only 11 states and transitions, the results. The authors do not attempt to interpret these numbers; they only serve as a demonstration of how Markov analysis can be used to study design protocols. One would expect it will take more steps to move from Function (F) to Structure (S) and fewer steps to move from expected Behaviour (Be) to Structure (S).

Statistical descriptions, cumulative occurrence, Markov chains and mean passage times provide quantitative models of design cognition—the cognitive behaviour of designers. They are used to gain insight into designing as a process. We now move on to how we can produce richer representations from protocol source data and how we can generate further quantitative models of design cognition that enhance our understanding of design. We can then use these models to examine similarities and differences in a large variety of design conditions.

3.5 Semantic Design Process

Semantic design processes are the design processes that are derived by considering the semantic linkage of design issues. After constructing the linkograph, if there are *n* links there will be *n* processes.

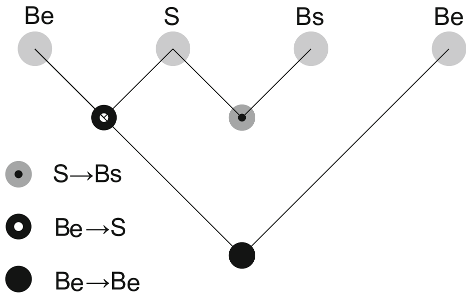


Fig. 3.11 Goldschmidt examples with coded FBS issues

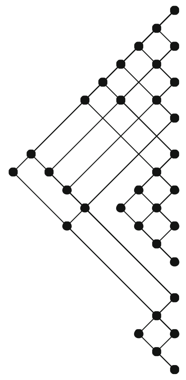
3.5.1 Deriving FBS Design Processes

Using Goldschmidt's example of four move in Sect. 3.4 by combining Fig. 3.6 and the FBS ontology codes in Sect. 3.4.1 we get Fig. 3.11, which shows the linkograph together with the FBS issues. In this example, three processes are derived. The link from move 1 to move 2 (Be → S) meets the definition of synthesis. The link from move 2 to move 3 (S → Bs) meets the definition of analysis. These agree with the understanding of the protocol. The last process, from move 1 to move 4 (Be → Bs), should not be classified as evaluation if we examine the design protocol. Move 4 is triggered by the "direction" aspect the designer discovers in move 1. If the behaviour code (B) is to be separated into expected behaviour (Be) and behaviour is derived from structure (Bs), the first move and the fourth move should be coded as Be, as both moves were anticipating the directional behaviour of the design. However, there is no process within the FBS framework to describe the (Be → Be) process, which can be viewed as a reflection process. In order to better capture design information from the design protocol, the situated FBS ontology will be used.

3.5.2 Deriving Situated FBS Design Processes

Since we have already subdivided the protocols using the situated FBS coding scheme we can construct a new linkograph by discerning the connections among the segments in Table 3.2. Table 3.4 shows the rotated linkograph in relation to the segments and situated FBS issues. The reasoning of the connections among segments is too lengthy to depict here. Interested readers can examine Table 3.4 to see if they agree with the authors' discernment. There are 23 links in the linkograph. If each link is considered as a transformation process, there will be 23 processes. If segment n is connected to segment (n + i), it is represented by n ⇝ (n + i) without specifying the type of processes as in Table 3.5. It also shows the frequencies and links of the derived processes. Observing the table, there are more structure-initiated processes than behaviour-initiated processes, 16 against 7.

Table 3.4 Example of coding with situated FBS



1	S^i	Structure before move 1
2	B^i	If I look at the form again it seems that spatially,
3	Be^i	These are the larger directions
4	S^e	Draw w x
5	Se^i	I am getting 1, 2, 3 spaces here and 1, 2 there
6	S^e	Draw p and q
7	S^i	They're about square
8	Se^i	So there is a tendency to try to see them as space
9	Be^i	These are secondary directions within the space
10	S^e	Draw y z
11	B^i	So the entry (3) is actually moving along the secondary directions

Table 3.5 Derived processes

Derived process	Frequency	Links between segments (segments inside brackets)
$B^i \rightsquigarrow Be^i$	2	(2 3) and (2 9)
$B^i \rightsquigarrow Se^i$	1	(2 5)
$Be^i \rightsquigarrow B^i$	1	(9 11)
$Be^i \rightsquigarrow Be^i$	1	(3 9)
$Be^i \rightsquigarrow S^e$	2	(3 4) and (9 10)
$S^i \rightsquigarrow B^i$	1	(1 2)
$S^i \rightsquigarrow Be^i$	2	(1 3) and (1 9)
$S^i \rightsquigarrow Se^i$	2	(1 5) and (7 8)
$S^i \rightsquigarrow S^e$	3	(1 4), (1 6), and (1 10)
$Se^i \rightsquigarrow S^i$	1	(5 7)
$Se^i \rightsquigarrow Se^i$	1	(5 8)
$Se^i \rightsquigarrow S^e$	1	(5 6)
$S^e \rightsquigarrow B^i$	1	(10 11)
$S^e \rightsquigarrow Be^i$	1	(4 9)
$S^e \rightsquigarrow S^i$	1	(6 7)
$S^e \rightsquigarrow Se^i$	1	(6 8)
$S^e \rightsquigarrow S^e$	1	(4 10)

It can be observed that there are 17 types of processes in this design session. Some of these derived processes map directly to the situated FBS processes. For example, in the first row of Table 3.10, the two $B^i \rightsquigarrow Be^i$ processes correspond to the focusing process ($B^i \Leftrightarrow Be^i$). In these two instances the designer focuses on the directional aspect (segment 2 to segments 3 and 9) of the spatial form.

However, some derived processes do not match the situated FBS processes. For example, in the fifth row of Table 3.5, the two $Be^{i} \rightsquigarrow Se$ processes cannot be found in Fig. 3.2. Revisiting the design protocol suggests these two occurrences of structure depiction (S^{e}) are depicting topological directions (*w, x, y* and *z*) and there should be a topological expectation of structure (Se^{i}) after the expected directional behaviour (Be^{i}). In this case, the $Be^{i} \rightsquigarrow S^{e}$ processes map to two situated FBS processes represented by $Be^{i} \rightarrow Se^{i} \rightarrow S^{e}$.

By re-visiting the protocol, codes are added in between the "mismatched" processes. For example, in a large number of cases the external structure code (S^{e}) requires a structure interpretation code (S^{i}). For examples in segment 11 the designer discovers that the entrance 3 is "moving along the secondary direction"; this is an analysis of the depicted structure of *y* and *z* (segment 10) and the entrance. Before s/he analyses the structure, an interpretation of the structure is required. In some cases two additional codes are required, for example, the link between the depicted structures ($S^{e} \rightsquigarrow S^{e}$) segment 4 and segment 10. In Segment 4 the designer draws the *w* and *x* axes and in segment 10 the designer draws the *y* and *z* axes. These two segments are linked because without drawing the major *x* and *w* directions, s/he might not discover the "secondary" directions *y* and *z*. In the situated FBS framework, this drawing action involves interpreting existing drawings, focusing on the directions and then expecting the "secondary" directions before drawing *y* and *z*. These three processes are represented by: $S^{e} \circlearrowleft S^{i} \Leftrightarrow Se^{i} \rightarrow S^{e}$.

Table 3.6 summarises the mapping of the derived processes to the situated FBS processes. Seven out of the 17 processes can be directly mapped to the situated FBS framework. Out of the 10 types of processes that cannot be directly mapped to the ontology, eight require one additional situated FBS state between the codes assigned to segments, two require two additional situated FBS states between the codes assigned to segments. These added codes can be seen as states (or cognitive activities) that are not directly observable.

3.6 Statistical Analysis

As seen in Fig. 3.7 and Table 3.5, using simple counting can help to compare and understand an episode. In this section we explore different statistical analysis methods to examine design protocols coded with the FBS ontology and linkography. Our goal is to provide a platform for quantitative descriptions of the cognitive activities related to designing.

3.6.1 Descriptive Statistics of Design Issues and Processes

Descriptive statistics of design issues can quantify the types of cognitive activities used during the episode. For example counting the different issues of Fig. 3.11 we

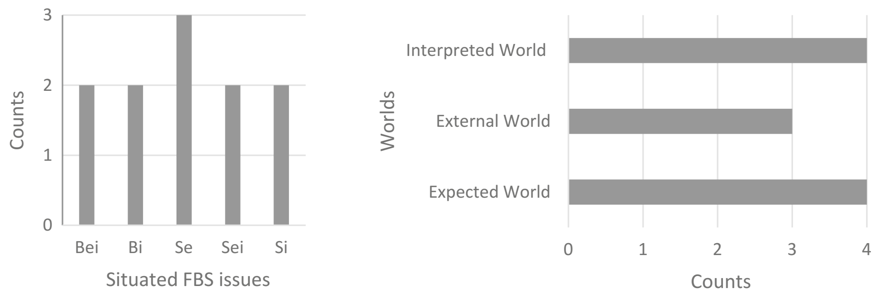
Table 3.6 Derived processes mapped to situated FBS processes

Derived process	Situated FBS process	Comments
B^iBe^i	$B^i \Leftrightarrow Be^i$	Focusing on the expected behaviour of "larger" and "secondary" axis, a kind of Reformulation II
$B^i\ Se^i$	$B^i \Leftrightarrow Be^i \rightarrow Se^i$	Expected behaviour added between behaviour "… it seems spatially" and expected structure "I am getting 1, 2, 3 spaces" to complete the synthesis process
$Be^i\ B^i$	$Be^i \leftrightarrow B^i$	Evaluation of axial behaviour of entrance 3
Be^i Bei	$Be^i \Leftrightarrow B^i \Leftrightarrow Be^i$	Interpreted behaviour of "direction" added between the expected behaviour of "larger" and "secondary" directions, a kind of Reformulation II
Be^i Se	$Be^i \rightarrow Sei \rightarrow S^e$	Expected structure added to complete the documentation process
$S^i\ B^i$	$Si \rightarrow B^i$	Analysis of structure
$S^i\ Be^i$	$S^i \rightarrow Bi \Leftrightarrow Be^i$	Interpreted behaviour of "direction" added before the expected behaviour of "larger" and "secondary" directions
S^i Sei	$S^i \Leftrightarrow Se^i$	Focusing on the shape and "square"
$S^i\ S^e$	$S^i \Leftrightarrow Se^i \rightarrow S^e$	Focusing on the directions and square space before depicting them
Sei Si	$Se^i \Leftrightarrow S^i$	Focusing on the interpretation of squares
Sei Se^i	$Se^i \Leftrightarrow S^i \Leftrightarrow Se^i$	Justifying interpreting the expected structure as expected spaces
Sei S^e	$Se^i \rightarrow S^e$	Documentation of the p and q spaces
$S^e\ B^i$	$S^e \circlearrowleft Si \rightarrow B^i$	Analysis of axis entrance 3 and y, interpreted structure added
$S^e\ Be^i$	$S^e \circlearrowleft Si \rightarrow B^i \Leftrightarrow Be^i$	Interpreted structure and interpreted behaviour of "direction" added before the expected behaviour of "larger" and "secondary" directions
$S^e\ S^i$	$S^e \circlearrowleft S^i$	Interpreting the p and q spaces as squares
$S^e\ Se^i$	$S^e \circlearrowleft S^i \Leftrightarrow Se^i$	Interpreted structures of space before expecting the structure as spaces, p and q
$S^e\ S^e$	$S^e \circlearrowleft S^i \Leftrightarrow Se^i \rightarrow S^e$	Interpreted x w direction and expected secondary direction before depicting y z

get Fig. 3.12. This can be understood as during this episode, 75 % of the cognitive effort was spent on behavioural issues and 25 % of the cognitive effort was spent on structure issues.

However, if using the finer grained situated FBS model to investigate the episode and counting the occurrences of the situated FBS issues, Fig. 3.13a can be obtained. 36 % of the cognitive effort were put into behaviour issues. This cognitive effort was divided equally between the interpreted world and the expected world. The remaining 64 % were of structure issues; three out of seven of those were external depiction. The remaining structure issues were again equally distributed between the interpreted world and the expected world. Figure 3.13b shows the

Fig. 3.12 Counting the FBS issues



(a) Counting the situated FBS issues

(b) Counting issues in different worlds

Fig. 3.13 Counting the situated FBS issues

count of design issues in relation to the three worlds. Descriptive statistical examination of a design episode with the situated FBS ontology will give addition insights into the cognitive effort expended in different areas.

Similarly, syntactic and semantic FBS design processes of a design session or a design episode can be counted. The application of descriptive statistics to protocols from experiments will be given in Chap. 4.

3.6.2 Statistical Inference of Design Protocol: p Value

Descriptive statistics summarize or quantitatively describe designing in terms of the FBS ontology. Inferential statistics can be used to test hypotheses about designing. To illustrate this we use a common statistical test called paired Student's *t* test. To put it into context, we construct an example from Bilda's (2006) mental imagery design experiment.

Student's t test is one of the most commonly used techniques for testing a hypothesis on the basis of a difference between sample means. A paired t test looks at the difference between paired values in two samples, takes into account the variation of values within each sample, and produces a single number. Statistical software reports results as a probability. This probability is called the p value. The p value is not produced directly by the t test, it is calculated in one further step, using the outcome of the t test. In another words, it determines a probability of the chance of the two populations are the same with respect to the variable tested. The p value gives a predictive answer to the question of how certain it is that the null hypothesis is true. The lower this value is, the less likely the difference is by chance. The p value helps one to decide whether or not to accept the null hypothesis. Typically in protocol analysis significance level of 0.05 is used as a cut-off point to reject the null hypothesis.

Back to our example, the idea behind Bilda's experiment is that when a designer does not have access to sketching in the early conceptual stage, it will affect both the design process and the design outcome. Design literature shows a common agreement that sketching is essential for conceptual designing. However, Toker (2003) documented that Frank Lloyd Wright could conceive and develop the entire design using imagery alone and produce an external representation at the end of the process. Blida's study aimed to investigate the effects of not having access to sketching, in the early conceptual design phase, on the cognitive behaviour of a designer. Data collection involved six expert architects working on two different design problems on the same site under two different conditions, one in which they were blindfolded and hence could not sketch (called the experiment condition) and one in which they were sketching (called the control condition). Further details of the experiment details will be given in Chap. 6.

Here, to illustrate the use of paired t test, we simplify the study by testing the hypothesis that the blindfolded sessions will have less cognitive activities after 20 min of the design session. The reasons behind this hypothesis are first of all, on average, our attention span is about 20 min and second the cognitive activities will slow down because of memory load and the unavailability of sketches to off load cognition. Table 3.7 shows the number of segments in the first 20 min and the rest of the session with respect to these two conditions. The assumption is the number of segment represents number of cognitive activities. The null hypothesis is there will

Table 3.7 Number of segments in the first 20 min and the rest of the session

	Blindfolded no. of segments			Sketch no. of segments		
	20 min	Rest	Total (45 min)	20 min	Rest	Total (45 min)
Architect 1	89	78	167	68	77	145
Architect 2	63	91	154	77	107	184
Architect 3	87	82	169	65	77	142
Architect 4	92	75	167	74	95	169
Architect 5	73	72	145	91	62	153
Architect 6	69	53	122	71	101	172

Table 3.8 Number of segments in the first 20 min and the rest of the session

	% segments BF last 25 min	% segments SK last 25 min
Architect 1	46.7	53.1
Architect 2	59.1	58.2
Architect 3	48.5	54.2
Architect 4	44.9	56.2
Architect 5	49.7	40.5
Architect 6	43.4	58.7
Average	48.7	53.5

be no difference in the percentage of cognitive activities of the two condition. Table 3.8 shows the percentages of segments in the last 25 min. Table 3.9 shows a typical summary of results when doing a *t* test on Table 3.8. The normal convention to report this results is: $t(5) = -1.34$, $p < 0.12$. Taking $p < 0.05$ to be the significance level, we cannot reject the null hypothesis, which is there is no difference in the percentage of cognitive activities of the two condition. Note that in this example we are using a number of participants that may be too small to produce statistical robustness, i.e., more reliable results will be produced with a larger sample size.

Does the result conclude there is no difference in the cognitive activities when designing blindfolded? No, the *t* test result suggest there is a difference of *t* value of −1.34, and at the probability of the difference being due to chance is less than 0.12, so we cannot be confident in rejecting the null hypothesis with this set of data. Any standard statistics textbook provides a detailed exposition of testing.

Table 3.9 Typical *t* test results

	% segments BF last 25 min	% segments SK last 25 min
Mean	48.72	53.48
Variance	30.97	45.06
Observations	6	6
Pearson correlation	−0.006013977	
Hypothesized mean difference	0	
df	5	
t Stat	−1.335499967	
P(T ≤ t) one-tail	0.119639738	
t Critical one-tail	2.015048373	
P(T ≤ t) two-tail	0.239279477	
t Critical two-tail	2.570581836	

3.6.3 Statistical Description of Linkographs

Classical protocol analysis uses statistics to measure segment categories. In linkography there is no categorical data. However, it can be observed in the graphs that some parts have a higher density of links than others. This section uses standard statistics and methods of clustering to describe a linkograph in such a way as provide information on which to base further insights into designing.

The linkograph in Fig. 3.8 can be re-represented by taking out all the linking lines, as in Fig. 3.14. Here the first move is assigned as the origin and there is a one-unit separation between each move. The position of each node (link) will have a coordinate in the X-Y plane. The linkograph can then be statistically described in terms of the total number of nodes and the statistical position of links, which are the mean values of $(\bar{x}, \bar{y})$ and their standard deviations (σx, σy). The total number of nodes indicates the level of saturation of a linkograph. Normalising this number against the number of moves will be the link index.

Table 3.10 and Fig. 3.15 show the statistics and scatter plot of the linkograph. A higher mean value of x, $\bar{x}$, implies that more links appear at the end of a session and a lower value suggests that more linked nodes are present at the beginning of the session. A higher mean value of y, $\bar{y}$, indicates longer linking lengths. However, the mean values do not include the dispersion of the distribution, therefore, standard deviations are measured to indicate how concentrated the nodes are clustered around the means. Tables 3.11 and 3.12 relate the appearance of linkographs, with the same number of links, to the statistical values of x and y respectively. The figures in the two tables are exaggerated for illustration. For this example, Fig. 3.12 and Table 3.10, there are more links towards the end of the session since $\bar{x}$ is greater than the median or middle point.

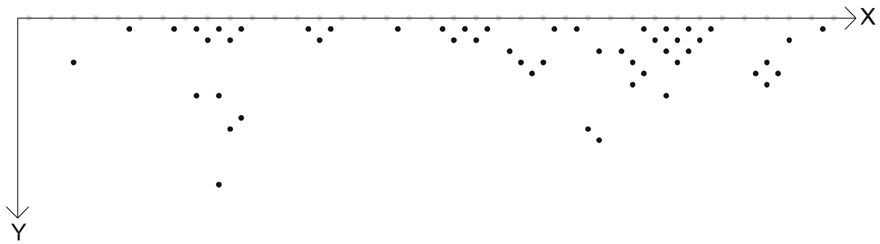


Fig. 3.14 Re-representation of the linkograph in Fig. 3.8 with nodes only in a 2-D space, the *lines* connecting the nodes to the moves have been removed, a *black dot* denotes a link between two moves

Table 3.10 Descriptive statistics of the example linkograph

	N	Minimum	Maximum	Mean $(\bar{x}, \bar{y})$	Std. deviation
X	52	3.00	36.50	22.01	9.41
Y	52	7.50	0.50	1.76	1.56

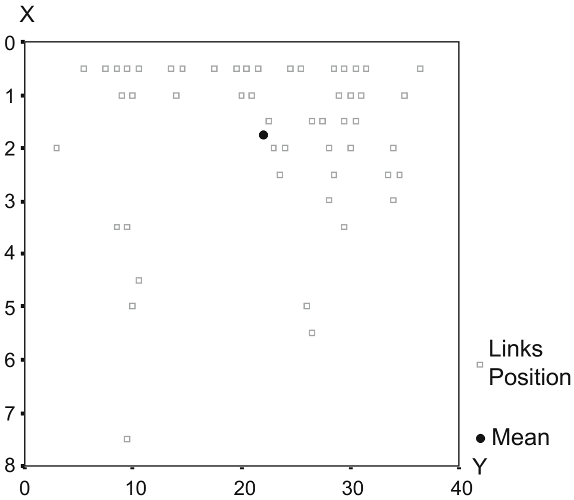


Fig. 3.15 Scatter plot of the linkograph with mean value

Table 3.11 Shape of a linkograph in relation to values of $\bar{x}$

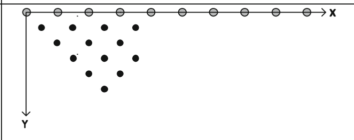
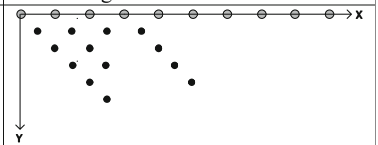
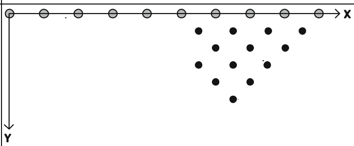
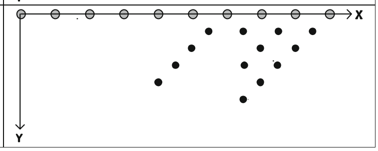
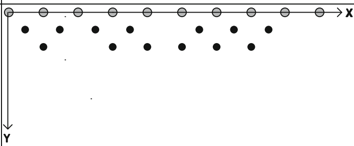
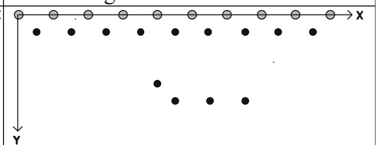
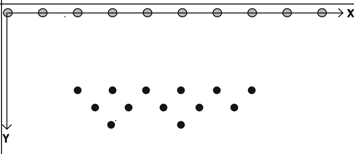
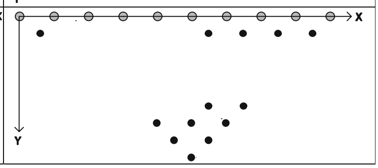
X Axis	Small standard deviation	Large standard deviation
Small mean, $\bar{x}$		
Large mean, $\bar{x}$		

Table 3.12 Shape of a linkograph in relation to values of $\bar{y}$

Y Axis	Small standard deviation	Large standard deviation
Small mean, $\bar{y}$		
Large mean, $\bar{y}$		

Cluster Analysis of Linkographs

Examining Fig. 3.8, there seem to be two chunks in this linkograph. The first chunk is from move 1 to move 18 and the second chunk from move 19 to 37. Comparing these two chunks in Fig. 3.8 to the scatter plot of Fig. 3.12, the links in a linkograph can be considered as data points that may form clusters in the x-y plane. These clusters resemble the chunks of ideas that are interlinked. Any clustering algorithm can be used to explore whether it is possible to cluster these two chunks automatically. This will complement the visual inspection to find the number of chunks and eliminate subjectivity. Most clustering algorithms can handle both continuous and categorical variables. The positions of links, those two-dimensional points (x, y) (nodes), are the data for clustering. In the first step of this procedure, the data are pre-clustered into many small sub-clusters, according to the selected criteria. Then, the algorithm clusters the sub-clusters that were created in the pre-cluster step into the desired number of clusters. If the desired number of clusters is unknown, the algorithm automatically finds the appropriate number of clusters according to the criteria. In this study the x and y variables were treated as continuous and Euclidean distance (the two-dimensional distance between links (x_i, y_i); and (x_j, y_j) computed by: $\sqrt{(x_i, y_i)^2 + (x_j, y_j)^2}$) was used to compute the distance among clusters. Akaike's information criterion (Akaike 1973), based on the maximum likelihood principle, was used for determining the number of clusters. Figure 3.16 shows the two groups of clusters found by the algorithm (here we used SPSS) which resemble the chunks. Table 3.13 shows the cluster distribution and Table 3.14 shows the cluster profile. From the profile we can deduce that Group 1 has longer links than Group 2 because of its higher $\bar{y}$ value; also the links in Group 1 are more scattered in the y directions because of a higher standard deviation.

From the distribution we can see that Group 2 (35) contains more than double the links in Group 1 (17). The link index for Group 1 is 17/18 (0.94) and the link

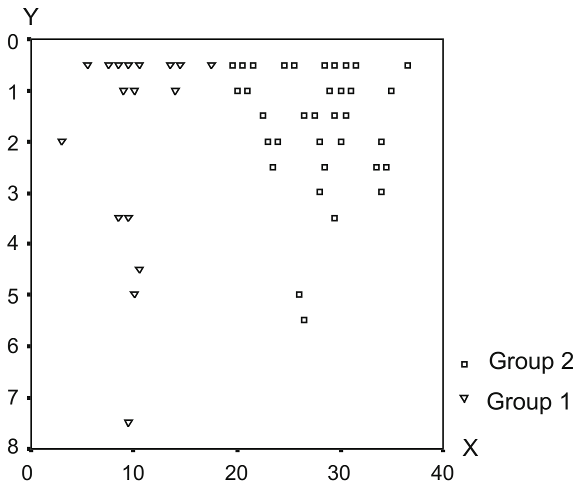


Fig. 3.16 Scatter plot of the two clusters generated by SPSS

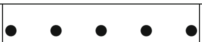
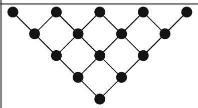
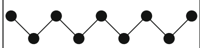
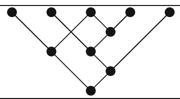
Table 3.13 Cluster distribution of the linkograph

	N	% of combined	% of total
Cluster			
Group 1	17	32.7	32.7
Group 2	35	67.3	67.3
Combined	52	100.0	100.0
Total	52		100.0

Table 3.14 Cluster profile of the linkograph

Centroids	X		Y	
	Mean ($\bar{x}$)	Std. deviation	Mean ($\bar{y}$)	Std. deviation
Cluster				
Group 1	10.06	3.42	1.94	2.11
Group 2	27.81	4.59	1.67	1.24
Combined	22.01	9.41	1.76	1.56

Table 3.15 Hypothetical linkographs of five design moves and their interpretations

Case 1		Five moves are totally unrelated, indicating no converging ideas, hence very low opportunity for idea development
Case 2		All moves are interconnected; this shows that this is a totally integrated process with no diversification, hinting that a premature crystallisation or fixation of one idea may have occurred, therefore there is a very low opportunity for novel ideas
Case 3		Moves are related only to directly preceding moves. This indicates the process is progressing but not developing, indicating some opportunities for idea development
Case 4		Moves are inter-related but not totally connected, indicating that there are lots of opportunities for good ideas with development

index for Group 2 is 35/19 (1.84). According to Goldschmidt (1992), the second half is more productive than the first half. The overall session link index is 52/37 (1.41). Essentially, the link index indicates the situation of a linkograph. From a theoretical viewpoint, is a saturated linkograph desirable? Does a fully linked linkograph indicate no diversification of ideas, hence less opportunity for creative outcome? This proposition is exaggerated with four hypothetical design scenarios in Table 3.15. We speculate that a partially linked linkograph embodies a balanced process in the sense that it embraces both integration and diversification of ideas. The figures in Table 3.15 suggest that the opportunity for idea development has some relationship with the predictability of the links in the linkograph. The links in Case 1 and Case 2 are predictable in the sense that they are either all linked or all unlinked. It is very easy to describe them. In Case 3 and Case 4 there are many more possibilities; more words are needed to describe them. The amount of words needed to communicate those linkographs directs the study to explore the use of the information theory of communication (Shannon 1948) to measure the graphs.

3.7 Information Theory

Shannon (1948), the founder of information theory, suggested that communication of information can be measured by the probability of its outcome and the semantics of information are irrelevant. The amount of information carried by a message or symbol is based on the probability of its occurrence. If the probability is 1, there is only one possible outcome, then there is no need to communicate additional information because the outcome is known. In the hypothetical cases in Table 3.15, there are ten possibilities of linkage. Cases 1 and 2 can be considered as all unlinked and all linked. Only one signal or symbol is needed to communicate them. In Cases 3 and 4 the probabilities of having a link are 4/10 and 5/10 respectively; more symbols are needed to communicate them. This section will propose how to use information theory to describe and measure a linkograph. It will start with the information-generation function and the calculation of entropy, which is the unit of measurement of information.

In Shannon's formulation of information theory, communication systems are modelled as a stochastic process (a simple definition of stochastic process is an ordered collection of random variables) of information transmitted from a source through a channel. Information is transmitted through recognisable symbols pre-determined by the source and the receiver (encoding and decoding). If the outcome is known then there is no additional information. To illustrate this with a simple example, consider transmitting a piece of information consisting of ten ON/OFF signals and one of them is OFF but the others are ON. The probability of an OFF symbol, p(OFF), is 0.1 and the probability of an ON symbol, p(ON), is 0.9. Consider the following two cases:

1. If the first signal the receiver gets is an OFF symbol ($p = 0.1$), then no further transmission is required as the following signals carry no additional information. This, a stochastic process, assumes that the receiver knows the total number of signals (10), the probabilities of the symbols (ON/OFF), and that the total probability equals 1 (p(ON) + p(OFF) = 1).

2. If the first signal being transmitted is an ON symbol ($p = 0.9$), then the receiver is uncertain of the value of the next signal. Further transmission is still required to complete the information.

The transmission of the first case carries more information. The amount of information carried by a symbol (ON or OFF in this case) is related to the probability of its outcome.

Another example concerns the game of bridge. If a player calls something that surprises her/his partner, her/his partner gets more information. Based on these kinds of observations, Shannon proposed an information-generating function *h(p)*. This information function needs to have the following properties:

h(p) is continuous for $0 \leq p \leq 1$, where p is the probability;
$h(p_i)$ = infinity if $p_i = 0$, where pi is the probability of a given state;
$h(p_i) = 0$ if $p_i = 1$;
$h(p_i) > h(p_j)$ if $p_j > p_i$, where p_i and p_j are the probabilities in two different states; and
$h(p_i) + h(p_j) = h(p_i \times p_j)$ if the two states are independent.

Shannon proved that the only function that satisfies the above five properties is:

$$h(p) = -\log(p) \tag{3.9}$$

Given a set of N independent states $a_1,\ldots a_n$ and the corresponding possibilities $p_1,\ldots p_n$, (in the above example, $N = 2$, $p_1 = p(\text{ON}) = 0.9$, and $p_2 = p(\text{OFF}) = 0.1$). Shannon derived entropy (H), the average information per symbol in a set of symbols, to be:

$$p_1 \times h(p_1) + p_2 \times h(p_2) + \cdots + p_n \times h(p_n) \tag{3.10}$$

Therefore

$$H = \sum_{i=1}^{n} p_i\{\log_b(p_i) \quad \text{with} \quad \sum_{i=1}^{n} p_i = 1 \tag{3.11}$$

In the example there are two symbols (ON/OFF) and entropy is expressed by:

$$H = -p(\text{ON})\log_2(p(\text{ON})) - p(\text{OFF})\log_2(p(\text{OFF})) \tag{3.12}$$

Substitute the values of probabilities:

$$H = -(0.9 \times \log_2(0.9) + 0.1 \times \log_2(0.1)) = 0.469 \tag{3.13}$$

The "logarithmic base corresponds to the choice of a unit for measuring information" (Shannon 1948, p. 379). Here base 2 is used to represent the binary (a binary system represents numeric values using two symbols, usually 0 and 1) ON and OFF information.

The next section describes how this can be applied to calculate the entropy of a linkograph of a design session.

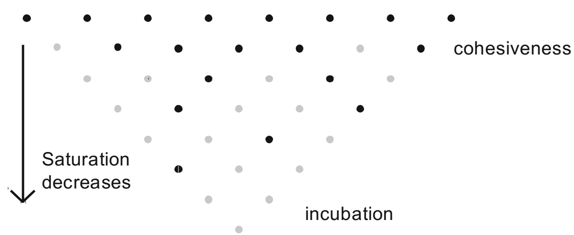
3.7.1 Entropic Measurement of Linkographs

The authors consider an empty linked linkograph as a non-converging process with no coherent ideas and that a fully linked linkograph stands for a wholly integrated process with no diversification (refer to Table 3.12). In both cases the opportunities for idea development are very low. This line of reasoning can be expressed in terms of entropy; if a move is randomly picked from an empty linked linkograph, we can be certain that it is not linked to any other moves. This sounds obvious, but this linkograph can be considered as a carrier with zero information content; because the outcome is known, it will have zero entropy. Similarly, a fully linked linkograph will also have zero entropy.

In order for the entropy measurement of a linkograph to be meaningful, the conceptual differences between forelink and backlink must be considered. A third link type called a horizonlink is introduced. A horizonlink is not a link itself but it bears the notion of length of the links, which also maps onto time (separation) between links. It can be viewed as a measure of the distances of the links. It characterises two opposite notions: cohesiveness and incubation. Figure 3.17, where black dots denote linkages between moves and grey dots denote no linkage between moves, shows a typical linkograph with more cohesive links (short links) than incubated links (long links). When considering the short links, if ideas are not cohesive there is a lack of integrations hence they are not desirable. However, if ideas are too cohesive there is a lack of innovation. Similarly, totally connected long links indicate lack of diversification. In practice, however, long links are rare and are usually desirable, as they revisit previous ideas, which might indicate the importance of those ideas. Figure 3.18 shows three abstracted linkographs for entropy measurement;.

In Fig. 3.18c, it can be observed there are $n - 1$ rows in an n moves linkograph. Let $n - i$ denotes the row number; the links in rows with a small i indicate that the distance between moves is small, and they are labelled as short links. These moves will likely reside in working memory and are referred to as the cohesiveness of

Fig. 3.17 A linkography with typical distribution of links during a design process



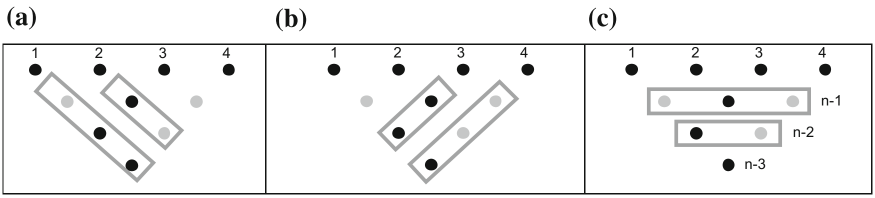


Fig. 3.18 Entropy measures of linkograph: **a** forelink, **b** backlink, and **c** horizonlink

ideas. However, if the ideas are too cohesive, that might imply fixation and lack of innovation. The links in the rows with a larger i connect moves that are far apart; they are called long links. These moves may not be in the working memory and are considered as incubated moves. Long links are comparatively rare and may signify reflection in action. The authors assume that a good design process is reflected in a linkograph that contains unsaturated short links (cohesive links) plus a number of long links (incubated links).

The forelink entropy for each move is computed by Eq. 3.12, except for the last two moves. The p(ON) represents the probability of linkage and p(OFF) represents the probability of no linkage. For the last two moves, as seen in Fig. 3.18a, move 4 will not have forelinks and move 3 is either linked or unlinked to move 4, which will have zero entropy. Similarly, each segment except the first two will receive a backlink entropy, Fig. 3.18b. The moves legitimate for entropy calculation are enclosed by rectangles in the figure. In move 1 there are three nodes for links inside the rectangle; move 1 and move 2 are unlinked, while move 1 is linked to move 3 and move 4. The percentage of linked nodes is 66.6 % and the percentage of unlinked nodes is 33.3 %. So the probability will be: $p(\text{ON}) = 0.666$ and $p(\text{OF F}) = 0.333$ respectively. If we substitute these in Eq. 3.12, the forelink entropy for move 1 becomes:

$$H = -0.666\log_2(0.666) - 0.333\log_2(0.33) = 0.918$$

Similarly, the forelink entropy for move 2:

$$H = -0.5\log_2(0.5) - 0.5\log_2(0.5) = 1$$

As for move 3, there is only one possible link. No matter whether it is ON or OFF, the probability is 1 and the entropy is zero, because $\log_2(1) = 0$.

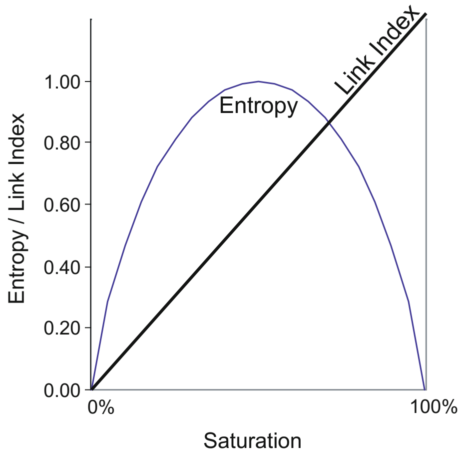
Using this method, the backlink entropies for move 3 and move 4 in Fig. 3.18b are 0 and 0.918 respectively.

For the horizonlink entropy in this case, only two rows are considered: $n - 1$ and $n - 2$. If those are computed with Eq. 3.12, the entropy of the $n - 1$ row is 0.918 and the entropy of the $n - 2$ row is 1. Since people have limited short-term memory (Miller 1956), applying Miller's "magic number seven plus or minus two" objects, linkographs seldom have segments with more than nine links and the number of links between far-apart segments will decrease. Figure 3.14 shows a typical linkograph which has many cohesive links but very few incubated links. A fully cohesive link, for example, all ON in $n - 1$, will have 0 horizonlink entropy; similarly, if there are no incubated links, that row will score 0 in horizonlink entropy as well.

If an idea is not used, it will not have many forelinks and this is represented by low entropy. However, if an idea has too many forelinks, this might indicate fixation; this is also indicated by low entropy. Backlink entropy measures the opportunities according to enhancements or responses. If an idea is very novel, it will not have backlinks. The resulting entropy is low. On the other hand, if an idea is backlinked to all previous ideas, it is not novel. Hence, it is represented by low entropy. Horizonlink entropy measures the occurrence of incubated segments. Low horizonlink entropy indicates complete cohesiveness. Horizonlink entropy measures the opportunities relating to cohesiveness and incubation.

The proposition that an intensively linked linkograph indicates good designs should apply up to a certain point of saturation. In the early stages of designing, fixation is not desirable. Fixation is indicated by a move with near- saturated forelinks. Here the suggestion is that forelink entropy measures the idea-generation opportunities in terms of new creations or initiations. Figure 3.19 compares the link index measurement with the entropy measurement of a move based on Eq. 3.12. A heavily linked and a sparsely linked linkograph will have low entropy values. However, the link index increases as the number of links increases. The slope of the link index in Fig. 3.19 is not fixed, as it is determined by the total number of moves of the linkograph. In this particular graph, it can be observed that the link index matches, but not closely, the entropy until the graphs intersect at about 75 % of saturation. It is very rare to have linkographs over 10 moves with that level of saturation. After this point, entropy drops, while the link index value continues to increase.

Fig. 3.19 Compare link index and entropy measurement



It can also be observed that the entropy curve in Fig. 3.19 is symmetrical; the slope of the graph decreases sharply as the probability moves away from 0 and 1. This indicates that when the links move away from determinate values of 0 and 1 (all un-linked and all linked), the H value increases rapidly. This graph shows that when $p(1)$ is between {0.35, 0.65}, H is over 0.93, that is, if the links in a row are between 35 and 65 %, it will produce a very positive value (rich design process). If the links are less than 5 % or over 95 %, it will produce a very low H value (below 0.29).

To illustrate the differences in these two measurements, link index and entropy, of linkographs, we use the hypothetical cases in Table 3.15 again. In Case 1 the probability of ON for all moves is 0 and the probability of OFF is 1, put these in Eq. 3.12, H = 0 because $\log_2(1) = 0$. Therefore the entropies will be 0 for any moves in any direction, hence the cumulative entropies will be 0.

For Case 2 the probability of ON for all moves is 1 but the probability of OFF is 0, so again, similar to Case 1, the cumulative entropies will be 0.

In Case 3, consider the forelink entropy of:

- the first move, there is one link out of four possible links, therefore the p(ON) = 1/4 = 0.25 and p(OFF) = 3/4 = 0.75, so H = 0.81;
- the second move, there is one link out of three possible links, p(ON) = 1/3 = 0.33 and p(OFF) = 2/3 = 0.67, so H = 0.92;
- the third move, there is one link out of two possible links, p(ON) = 0.50 and p(OFF) = 0.50, so H = 1.00;
- the fourth move only has one possible link, so no matter it is ON or OFF entropy value will be zero;
- the fifth move does not have any forelinks, so no entropy value.

The cumulative forelink entropy will be 0.81 + 0.92 + 1 = 2.73. As for the backlink cumulative entropy, the calculation will be similar to that of the cumulative forelink entropy but in the reverse order and with the same values.

The horizonlink entropy will be calculate by rows; starting from the bottom, the first row has only one possibility of ON and OFF so the entropy is zero. The second row has two possible links and both are OFF, so the entropy is zero again; this is the same for the third row. As of the fourth row, there are four possible ON and OFF links, in the case all are ON and the entropy is zero. Therefore the cumulative horizonlink entropy is zero.

In Case 4, consider the forelink entropy of:

- the first move, there are two links out of four possible links, therefore the p(ON) = 2/4 = 0.50 and p(OFF) = 2/4 = 0.50, so H = 1.00;
- the second move, there are one links out of three possible links, p(ON) = 2/3 = 0.67 and p(OFF) = 1/3 = 0.33, so H = 0.92;
- the third move, there is one link out of two possible links, p(ON) = 0.50 and p(OFF) = 0.50, so H = 1.00;
- the fourth and fifth moves have no entropy value.

The cumulative forelink entropy will be 1.00 + 0.92 + 1.00 = 2.92.

Consider the backlink entropy of:

- the first move, does not have any backlink, so no entropy value;
- the second move has only one possibility for link, so entropy will be zero;
- the third move, there is one link out of two possible links, so p(ON) = 0.50, p(OFF) = 0.50 and H = 1.00;
- the forth move, there are two links out of three possible links, p(ON) = 2/3 = 0.67 and p(OFF) = 1/3 = 0.33, so H = 0.92;
- the fifth move, there are one links out of four possible links, therefore the p(ON) = 2/4 = 0.50 and p(OFF) = 2/4 = 0.50, so H = 1.00.

Therefore the cumulative backlink entropy for Case 4 is 1.00 + 0.92 + 1.00 = 2.92.

Consider the horizonlink entropy of:

- the bottom row, which has only one possible links, therefore no matter it is linked or not linked the entropy will be zero;
- the second row, there is one link out of the two possible links, so p(ON) = 0.50, p(OFF) = 0.50 and H = 1.00;
- the third row, there are two links out of three possible links, p(ON) = 2/3 = 0.67 and p(OFF) = 1/3 = 0.33, so H = 0.92;
- the fourth row, there is one link out of four possible links, therefore the p(ON) = 1/4 = 0.25 and p(OFF) = 3/4 = 0.75, so H = 0.81.

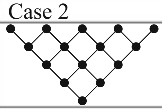
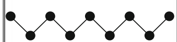
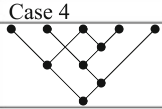
Therefore the cumulative horizonlink entropy for Case 4 is 1.00 + 0.92 + 0.81 = 2.73.

The link index of Case 1 equal zero because there is no links. There are 10 links in Case 2 so the link index is 2 (10/5, i.e. 4 links divided by five moves).

The link index of Case 3 equals 4/5 (4 links divided by 5 moves) and the link index of Case 4 is 5/5 = 1.

Table 3.16 compares the link index values and the entropy values of the hypothetical case depicted in Table 3.15. The entropy values in the table are the

Table 3.16 Comparison of the cumulative entropy and link index of hypothetical case

	Case 1	Case 2	Case 3	Case 4
				
Forelink H	0.00	0.00	2.73	2.92
Backlink H	0.00	0.00	2.73	2.92
Horizonlink H	0.00	0.00	0.00	2.73
Total H	0.00	0.00	5.46	8.55
Link index	0.00	2.00	0.8 0	1.00

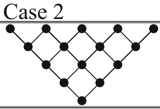
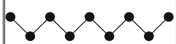
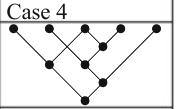
cumulative values of the contribution of each moves. The total value is the addition of the three different types of entropy. Link index benchmark Case 2 is the most productive scenario. However, as explained in Table 3.15, this might not be the most desirable scenario.

3.7.2 Normalizing Entropic Measurement for Comparison

Table 3.16 shows the cumulative values of entropy, which will increase as the number of moves increase. It is possible to calculate the maximum entropy of a linkograph and normalize against it by dividing the entropy calculated by the maximum entropy. This will always give us a number less than or equal to one. In our calculation, the maximum entropy occurs when the probability of link and unlink have the same value. i.e. when $p(\text{ON}) = p(\text{OFF}) = 0.5$, H = 1. This happens when there are even numbers of possible links, as we can observe only about half of the graph will have this maximum entropy of 1 and the remaining moves will always have an entropy of less than 1. So for a graph with n moves the maximum forelink and backlink entropy are (n − 1) and the maximum horizonlink is (n − 2). Table 3.17 shows the normalized entropy of the example in Table 3.15.

In summary, this section proposes using entropy to measure and study linkographs, in addition to link index and critical move analysis. Also, it describes how to calculate the entropy of a linkograph. The contribution of each move is counted in three different ways: according to forelinks, backlinks and horizonlinks. It is hypothesized that entropy measures the idea development opportunities. Forelink entropy measures the idea-generation opportunities in terms of new creations or initiations. Backlink entropy measures the opportunities according to enhancements or responses. Horizonlink entropy measures the opportunities relating to cohesiveness and incubation. Further, it is hypothesised that the entropy measurement of a linkograph is positively correlated to the design outcome, due to better opportunities for idea development.

Table 3.17 Normalized entropy of hypothetical case

	Case 1	Case 2	Case 3	Case 4
				
Forelink H	0.00	0.00	0.68	0.73
Backlink H	0.00	0.00	0.68	0.73
Horizonlink H	0.00	0.00	0.00	0.91

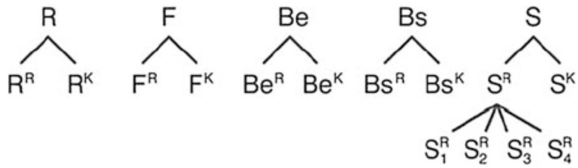


Fig. 3.20 Expanding FBS scheme to cover subclass variables (after Yu et al. 2013)

3.8 Summary

This chapter has proposed an ontological coding scheme and described its application. As an example of how this coding scheme can be applied more widely, Yu et al. (2013) utilised the FBS coding scheme to study designer's behaviour in a parametric design environment by instantiating the codes into multiple subclasses, Fig. 3.20. This does not modify the ontology as the subclasses with their own codes can be aggregated back into the FBS primary codes. They defined two types of design spaces: design knowledge space (denoted by the superscript K) and rule algorithm space (denoted by the superscript R). In the design knowledge space designers make use of their design knowledge and in the rule algorithm space designers apply design knowledge through the operations of parametric design tools. The structure variables in the rule algorithm can have more subclasses of the specific rule algorithm activities in the parametric design environment. By doing this, distinct activities can be mapped back to the FBS class variable and comparisons can be made with other design situations.

This chapter also revisited the linkography technique. Syntactic and semantic design processes have been defined. Statistical analysis methods have been depicted as a means to a describe design session and to produce inference-based observations. Markov chains have been proposed to study the design protocol as a time series, which provides another venue for examining design protocol data. In addition, two mathematically based methods, statistical clustering and Shannon's entropy, were proposed to analyse linkographs. This chapter has covered the theoretical background of using these quantitative methods in addition to those traditional methods of linkography to investigate design protocols.