

# Contents and Structure in Design Reasoning

Gabriela Goldschmidt and Maya Weil

## 1 Introduction: Design thinking and reasoning

"Design is based on acquiring skills, practice and, experience. Now, it is understood as an outcome of thinking processes."<sup>1</sup> This statement reflects a growing interest in gaining a better understanding of the ways in which designers think, on the part of both the design community and the cognitive science community. Different investigators have different objectives and different motivations in pursuing design thinking research. Some of the objectives and motivations are practical—the wish to develop design support systems, computational for the most part, that would hopefully be useful in early phases of conceptual design. Existing tools have, hitherto, offered little assistance in early design phases, and it is hoped that a better understanding of the ways in which the human mind functions with regard to design may afford a technological breakthrough. Other research orientations are more theoretical, and aim at explicating various aspects of human cognition—in this case design cognition, which is regarded as a prime example of creative thinking and problem solving.

Our research belongs to the second type. The present study focuses on design reasoning at the early, "front edge" phase of designing in which a solution to a design problem is sought at a preliminary, sketchy level. In this phase, designers are engaged in a search in which many ideas and possibilities are typically raised and examined, often to be discarded later. This is done by reasoning about bits of information and knowledge, in an attempt to construct a coherent rationale for a design idea, and later, a design proposal. We classify design problems as "ill-defined," by which we mean that there are no algorithms for their solution. This is what makes them so fascinating, and their solutions potentially creative, or at least innovative. At the same time, it makes it difficult to model the cognitive processes involved. The present study investigates the relationship—assuming there is one—between *contents* and *structure* in design reasoning. By *contents*, we mean the context-bound subject matters that are being considered, such as rooms and walls in the case of architectural design, or metal tubes and plastic casings in the case of industrial design. By *structure*, we mean consistent internal relationships and patterns that can be detected in the cognitive operations involved in design thinking. In order to examine a postulated relationship between the two, we must establish a common

---

1 From leaflet for "Design Odyssey 2010"—25th jubilee international symposium, Industrial Design Center, Indian Institute of Technology, Bombay, December 1994.

base that enables such an examination. We propose an investigation of online records of design behavior, parsed into individual design moves. Moves and relationships among them can be categorized in ways that pertain to both structure and content, making it possible to detect relationships and even to quantify them.

After briefly describing the experimental case on which we base our analyses in this study, we discuss our methodology, which is frequently used for design thinking research, namely verbal report (think-aloud or conversation) protocol analysis. We then present a particular analytical tool that is used in this study, linkography. Next, we report the results of the analytic work and, finally, we discuss our findings in terms of our topic: the relationship between contents and structure in design reasoning.

## 2 The design episode

The protocol used in this study records a design session in which a team of three industrial designers worked on the conceptual design of a bicycle carrier for a backpack.<sup>2</sup> The work lasted for two hours. The designers inspected the bicycle and the backpack in question, and were provided with technical drawings and other relevant information. They used a whiteboard for writing, and made sketches on four sheets of paper. The printed protocol includes this graphic output in addition to the transcript of their oral discourse. The current study focuses on a portion of the protocol, lasting 21 minutes (from 1 hour 17 minutes to 1 hour 38 minutes). This portion of the protocol was previously analyzed by Goldschmidt,<sup>3</sup> using linkography (see section 4). In the present study, we use data collected for that study (although the purposes of the current and the previous studies are entirely different). The selected portion of the protocol is divided into five units (out of 45 units in the entire protocol), comprising 298 moves.

The design phase under investigation is marked by intensive design activities. At this point, the team had already developed design criteria and listed a number of alternative concepts for the design of the carrier. It decides on a leading concept—the tray—and develops it and its implications for a comprehensive solution. The five design units deal with the following issues:

Unit 32: Tray and fastening devices

Unit 33: Tray features

Units 34/35: Mounting (human factors), and mounting-points

Unit 36: Features of (braze-on) joining

Unit 37: Complete rack and joints

The advantage of choosing this portion lies in the fact that it is a “middle of the road” phase that is neither concerned with preliminary concepts only, nor is it dedicated entirely to detailing and the refinement of concepts.

2 The protocol was generated by the faculty of Industrial Design Engineering, Delft University of Technology, for a workshop on Analyzing Design Activity held in Delft in September 1994. The papers presented in the workshop are assembled in Cross et al. (1996). A selection of these papers was published in *Design Studies* 16: 2 (1995).

3 G. Goldschmidt, “The Designer as a Team of One,” in N. Cross, H. Christiaans, and K. Dorst, eds., *Analysing Design Activity* (Chichester: John Wiley, 1996), 65–91. First published in *Design Studies* 16: 2 (1995): 189–210 (abridged).

### 3 Protocol analysis as a design thinking research methodology

The publication of *Protocol Analysis: Verbal Reports as Data* by Ericsson and Simon<sup>4</sup> marked a general approval of protocol analysis as a research methodology for the study of cognitive processes in many areas. Starting with work by Eastman,<sup>5</sup> a growing number of students of design thinking and behavior used protocol analysis as a means to get as close as possible to what actually takes place in the mind while designing. A concise review of the history of protocol analysis in design can be found in Cross, Christiaans, and Dorst,<sup>6</sup> who stress that "of all the empirical, observational research methods for the analysis of design activity, protocol analysis is the one that has received the most use and attention in recent years. It has become regarded as the most likely method (perhaps the only method) to bring out into the open the somewhat mysterious cognitive abilities of designers."

In design as in other fields, most protocol analysis studies are based on "think aloud" experiments in which individuals engage in the investigated activity while they continuously verbalize their thoughts. Protocols of these verbalizations are then parsed into small units and encoded using a category scheme that reflects the research objectives. Ordinarily parsing is straightforward. This is particularly true when the experiment involves more than one individual, in which case the protocol consists of natural conversation (as is the case in our example). Parsing often is carried out simply on the basis of time-units (for example, 3-minute units), but other parsing principles also have been used: for example, coherent arguments,<sup>7</sup> or simply natural utterances.<sup>8</sup> The coding scheme, on the other hand, may raise serious problems. Of importance are both the nature of the categories and their number. Ericsson and Simon,<sup>9</sup> referring to a study by Web,<sup>10</sup> comment: "An initial coding scheme, using a large number of different categories, was rejected because it made coding too difficult and unreliable. A second and much simpler scheme differentiated between three types of problem solving activities: preparation, production, and evaluation." Developing and fine-tuning an appropriate coding scheme is no trivial matter, as cogently pointed out by Purcell et al.<sup>11</sup> and by Gero and McNeill.<sup>12</sup> Coding schemes often are domain-specific and goal oriented; they depend on the objectives of the study and on the desired grain of analysis (see note 22 below). As we shall see, the question of coding is central to the topic of the present study.

Despite the recognized potential of protocol analysis to help in unveiling the "somewhat mysterious cognitive abilities of designers,"<sup>13</sup> it is met by some critics with much skepticism. It is not our intention to evaluate protocol analysis or to take issue with those who reject the methodology as such. Rather, we shall briefly present major criticisms and explain why we insist, nonetheless, on the benefits of using protocol analysis when appropriate:

- 4 K.A. Ericsson and H.A. Simon, *Protocol Analysis: Verbal Reports as Data* (Cambridge, MA: MIT Press, 1984/1993), 198; 204–5; 206–7.
- 5 C.M. Eastman, "On the Analysis of Intuitive Design Processes," in G.T. Moore, ed., *Emerging Methods in Environmental Design and Planning* (Cambridge, MA: MIT Press, 1970), 21–37.
- 6 N. Cross, H. Christiaans, and K. Dorst, "Introduction: The Delft Protocols Workshop" in N. Cross, H. Christiaans and K. Dorst, eds., *Analysing Design Activity* (Chichester: John Wiley, 1996), 1.
- 7 Ö. Akin, *Psychology of Architectural Design* (London: Pion, 1986).
- 8 P. Lloyd and P. Scott, "Discovering the Design Problem," *Design Studies* 15: 2 (1994): 125–140.
- 9 Ericsson and Simon, *Protocol Analysis*, 206–207.
- 10 N.L. Web, "An Exploration of Mathematical Problem-solving Processes" (Doctoral dissertation, Stanford University, 1975). *Dissertation Abstracts International*, 36, 2689A (University Microfilms No. 75-25625).
- 11 A.T. Purcell, J.S. Gero, H.M. Edwards, and T. McNeill, "The Data in Design Protocols: The Issue of Data Coding," in N. Cross, H. Christiaans, and K. Dorst, eds., *Analysing Design Activity* (Chichester: John Wiley, 1996), 225–252.
- 12 J.S. Gero and T. McNeill, "An Approach to the Analysis of Design Protocols," *Design Studies* 19: 1 (1998): 21–61.
- 13 Cross, Christiaans and Dorst, "Introduction."

- 1 *Thinking aloud alters regular cognitive processes, which are, therefore, not faithfully mirrored in think-aloud recordings.*

This point is debated in the literature, with pro and con evidence. There appear to be considerable individual differences among subjects, and protocols should only be analyzed if subjects do not report serious interference with their habitual processes.

- 2 *Cognitive operations cannot be reflected in verbal utterances because of differences in modality and speed.*

Whereas this is definitely true at least to some extent, concurrent verbalization comes as close to reflecting cognitive processes as we can possibly hope for at present, given available monitoring technologies. In design and related fields, an additional difficulty is caused by the fact that problem solvers also produce nonverbal output such as sketches and to date, no method has been proposed for their analysis under the same paradigm as verbal output. Accepting these limitations, one can still infer a wealth of information that is not accessible otherwise through the use of protocol analysis.

- 3 *It is only practical to record relatively short durations of work (a few hours at most). Designing is a process that evolves slowly over periods of time that may be quite long. Therefore, the process of designing cannot be captured using protocol analysis.*

This is a valid criticism and the methodology should, therefore, be used for studies of parameters that come to play in short stretches of time.

- 4 *Protocol analysis is a labor-intensive research methodology and, therefore, it is not suitable for large quantities of data. Any results that are obtained are, therefore, not generalizable.*

This, too, is a valid criticism, and it can be answered, as in the previous point, by suggesting that the methodology be used only for cases where large databases are not required. Where it is desirable to analyze larger bodies of data, or where protocol-analysis techniques offer no clear advantage, other methods should be used. For example, *replication protocol analysis*, developed by Galle and Kovács<sup>14</sup> for the purpose of studying decision justification, elicitation of knowledge, and evolution in design.

- 5 *Inter-rater reliability concerning coding is typically low.*

This difficulty may be related to the category scheme used, as pointed out by Ericsson and Simon,<sup>15</sup> or it may be the result of differences in the perception of raters who do not share the same microculture, and have not had extensive training.<sup>16</sup> Experimental psychology is well familiar with this problem, which is by no means exclusive to protocol analysis. It is, sometimes, solved by easing requirements regarding the number of raters (this has been our attitude in this study; see note 33).

To summarize, it is fair to say that protocol analysis is a methodology with clear limitations. However, no research methodology is "universal" in the sense that it is suitable for every purpose

14 P. Galle and L.B. Kovács, "Replication Protocol Analysis: A Method for the Study of Real-World Design Thinking," *Design Studies* 17: 2 (1996): 181–200.

15 Ericsson and Simon, *Protocol Analysis*.

16 Lloyd and Scott, "Discovering the Design Problem."

and under all circumstances. We hope to demonstrate in the present study that for specific, limited purposes, protocol analysis is not only an acceptable tool, but one that has considerable advantages if judiciously utilized.

#### 4 Linkography

As mentioned above, parsing and coding of protocols are of major importance to fruitful analyses. Parsing that is based on time-units, as is often the case (e.g., Eckersley;<sup>17</sup> Günther, Frankenberger, and Auer;<sup>18</sup> Lloyd and Scott;<sup>19</sup> Mazijoglou, Scrivener, and Clark;<sup>20</sup> and Radcliffe<sup>21</sup>) provides a rough division into phases of the investigated process. This is most useful when we are interested in the sequence of activities or topics addressed. For example, an account of instances in which shifts in activities or topics occur may teach us a lot about design behavior. A case in point is an analysis by Cross,<sup>22</sup> who was interested in capturing "the creative leap" in a design episode. He showed that different analyses of the protocol of that episode which are parsed into time units ranging from ten to twenty minutes, all reflect the instance of the "creative leap": they do so by recording a sharp shift in the activities, work loci, or discourse production, all category-sets that were used in coding the protocol segments in the different analyses.

While it is useful to look at time-dependencies of sets of variables, we believe that, by changing the base of parsing, protocols may be analyzed in additional ways, yielding rich and potentially illuminating information. *Linkography* is a system that parses protocols into individual *design moves*, independent of any time units.<sup>23</sup> Having established moves, the system records links among them. On the basis of patterns of links, it is possible to make inferences regarding different aspects of design reasoning and, through them, also assess design productivity. An introduction and detailed discussion of linkography and its utilization in assessing design productivity is undertaken elsewhere.<sup>24, 25, 26</sup>

##### 4.1 Links based on common sense

Linkography calls for the division of a protocol into units (by subject matter), which are then parsed into design moves. Typically, a unit comprises between twenty and one hundred moves. A move can be comprehended as follows: "The meaning of "move" in designing is akin to its meaning in chess: a design move is a step, an act, an operation, which transforms the design situation relative to the state in which it was prior to that move. Moves are normally small steps, and it is not always easy to delimit a move in the think-aloud protocol of a single designer... [a] team's protocol is easier to parse, and each utterance by one of the designers is defined as one move. Within each unit of the design process, the moves are numbered chronologically..."<sup>27</sup>

- 17 M. Eckersley, "The Form of Design Processes: A Protocol Analysis Study," *Design Studies* 9: 2 (1988): 86–94.
- 18 J. Günther, E. Frankenberger, and P. Auer, "Investigation of Individual and Team Design Processes" in N. Cross, H. Christiaans, and K. Dorst, eds., *Analysing Design Activity* (Chichester: John Wiley, 1996), 117–132.
- 19 Lloyd and Scott, "Discovering The Design Problem."
- 20 M. Mazijoglou, S. Scrivener, and S. Clark, "Representing Design Workspace Activity," in N. Cross, H. Christiaans, and K. Dorst, eds., *Analysing Design Activity* (Chichester: John Wiley, 1996), 389–416.
- 21 D.F. Radcliffe, "Concurrency of Actions, Ideas and Knowledge Displays Within a Design Team," in N. Cross, H. Christiaans, and K. Dorst, eds., *Analysing Design Activity* (Chichester: John Wiley, 1996), 343–364.
- 22 N. Cross, "Creativity in Design: Analyzing and Modeling the Creative Leap," *Leonardo* 30: 4 (1997): 311–317.
- 23 In the protocol segment we used in this study, 298 moves were made in 21 minutes. The average duration of a move is, therefore, 4.3 seconds. Consequently, the grain of an analysis based on moves is of a totally different order of magnitude than in an analysis based on time units lasting 20, 10, or even 3 minutes.
- 24 G. Goldschmidt, "Linkography: Assessing Design Productivity," in R. Trappl, ed., *Cybernetics and Systems '90* (Singapore: World Scientific, 1990), 291–298.
- 25 G. Goldschmidt, "Criteria for Design Evaluation: A Process Oriented Paradigm" in Y.E. Kalay, ed., *Evaluating and Predicting Design Performance* (Chichester: John Wiley, 1992), 67–79.
- 26 Goldschmidt, "The Designer as a Team of One."
- 27 Ibid., 72.

We focus our attention on the *links* that moves form among themselves, because we postulate that they are the key to the understanding of design reasoning. We are led to this conclusion by the great importance that we attach to the difference between well-structured and ill-structured problems. The former are solved using algorithms of one kind or another, but the latter cannot rely on algorithms. Therefore, a search is necessary, and in this search (which we shall not dwell on here) information is evoked and generated such that a rationale for a solution is produced. A solution and its rationale are constructed of many small fragments of information (their size is a matter of definition, relative to the grain of the investigator's analysis) that must be in agreement with one another. Moves are the representations of information made during the search and, in linking them to one another, the designer tests whether they are in agreement, thus reasoning about them. Bearing in mind how little we know about this ill-structured process, we believe that a study of the characteristics of links among design moves is essential to the understanding of design reasoning.

Links are determined solely on the basis of common sense, in a non-categorical manner: neither moves nor links are encoded. We understand common sense to mean "good sound ordinary sense" (*Webster's Third International Dictionary*). In practice, a link between two moves is established when the two moves pertain to the same, or closely related, subject matter(s), such as a particular component of the designed entity, its properties and functions, a concept or a design strategy, and so on. The linkography technique has each move, in chronological order, inspected for the existence or lack of existence of a link between it and any one of the previous moves (within a particular unit). It is, therefore, a binary system in which only "yes" and "no" answers are given to the question: is there a link between move *n* and move *n-k*. These links are called *backlinks* (notated <), because they go back in time. When the recording of backlinks in a given unit has been completed, we also can establish, after the fact, *forelinks* (notated >) that moves form: those are the links a move makes to subsequent moves. To illustrate the concepts of *move* and *link* (backwards and forwards), let us look at an example. We choose one move (move 26 in unit 32), which is given in Table 1 together with four moves it links back to, and two moves to which it has forelinks, within the same unit (please ignore the columns "C.S.1" and "C.S.2" for the time being). Move 26 addresses straps which are one of the options for connecting the given backpack to the bicycle rack that is being designed. All linked moves also deal with straps and with the functions they fulfill. We note that linked moves are not necessarily adjacent—a move may link to another move that is chronologically quite remote.



Table 1

**Common sense links and move categorizations (move 26, unit 32)**

Links	Move	Protocol	C.S.1	C.S.2
	26	well we sorta have that in the straps	A3	II
Backlinks	25	that's good for cinching down the load yeah	A3	II
	20	straps with snaps	B	III
	18	compression straps that also snap	A1	III
	16	cos then you could cinch your backpack down too you know nice compression straps	A1, A3	II
Forelinks	27	so do the straps	A1	III
	52	so we I think the issue that we're talking about is straps so we'll just keep that one on the burner	0	III, 0

Links are indicative of design productivity: the higher the links-per-moves ratio (called Link Index), the higher the productivity in a given unit, or design episode. Using a graphic notation system, we observe that links, by virtue of their quantity and "location" (regulated by the "distance" between each pair of linked moves), generate a pattern. This pattern is suggestive of certain reasoning characteristics of the process that is being investigated.<sup>28, 29</sup> For example, we observe "chunks" of links (a large number of links formed among a relatively small number of successive moves) at instances of increased design activity. Important moves tend to be situated at the beginning or the end of a sequence of moves that form chunks (we shall articulate shortly how important moves are identified). For our purpose here, the point we wish to stress is that the link-pattern reveals the structure of reasoning, whereas the links, themselves, because they are determined by common sense, are largely based on the contents of the moves that they associate.

#### 4.2 Critical Moves

The number of links that moves form is not invariable: some moves generate more links than others. We can count the links that each move makes, both backwards and forwards (back links can be counted as the analysis of the protocol progresses; forelinks can be summed up only when the analysis has been completed). Moves that generate a notably high number of links are of particular interest: we call them *critical moves* (CMs) and we postulate that they are more important than other moves in terms of advancing the design process, i.e., its productivity. Cross<sup>30</sup> has shown that critical moves, as determined by linkography, do indeed identify moves that, with hindsight, can be shown to be "breakthrough" moves in terms of concept-formation. For the purpose of analysis, we determine a threshold level of links per move: to be critical, a move must have at least the number of links indicated by the threshold level, either backwards or forwards. This level is flexible, and is fixed according to the grain of analysis and the purpose of the study. In the present investigation, this level is 6, implying that each critical move has 6

28 Goldschmidt, "Linkography."

29 Goldschmidt, "Criteria for Design Evaluation."

30 Cross, "Creativity in Design."

or more backlinks or forelinks (CM<sup>6</sup>). In addition, it may have links in the other direction; in fact, most moves generate links in both directions (see the example in Table 1). In rare cases, moves generate 6 or more links in both the backward direction and the forward direction.

We differentiate among moves that are critical due to a large number of backlinks (<CM), a large number of forelinks (CM>), or a large number of links in both directions (<CM>). Goldschmidt<sup>31</sup> has suggested that CM's> represent the creativity component of design productivity, but for creativity to manifest itself the design process must build, in equal proportion, on <CMS, which represent knowledge-based grounding of ideas. However, we must remember that, by and large, all critical moves form links in both directions, although they are classified according to where the larger number of links is to be found. Table 2 presents the distribution of back and fore links of CM's in the design episode under scrutiny. It shows that on the average, 73.2% of the links formed by a critical move are in the direction of its classification, and the rest of its links are in the opposite direction.

Table 2  
Back and Fore Links Formed By <CMs and CMs>

Unit	<CM			CM>		
	Quantity	<	Links >	Quantity	<	Links >
32	8	69	16	11	27	102
33	0	0	0	2	1	13
34/5	10	68	31	12	34	81
36	10	69	32	9	32	57
37	7	47	12	7	23	61
Total	35	253	91	41	117	314
%		73.5%	26.5%		27.2%	72.8%

Note: <CMs> are counted twice.

In the sections that follow, we shall concentrate on critical moves. They were determined by virtue of the links they form, based on their content: content leads to a common sense judgment regarding the existence of a link or its absence. The moves themselves were not encoded, and no system of categories was imposed on them. Our next task is to establish category schemes and encode the moves, disregarding their links. If the category schemes we apply pertain to structural properties of design activities, we should be able to correlate data regarding structure with data regarding contents, as reflected by analyses based on linkography.

31 Goldschmidt, "The Designer as a Team of One."



## 5 Structural analyses

We return to our parsed protocol now, this time attempting to encode the design moves for the purpose of further analysis. Our first task is to establish one or more coding schemes: there are no standard coding schemes, and researchers develop them ad hoc to fit their needs. We already commented on difficulties that may be caused by too large a number of categories in a coding scheme, or by lack of universality as a result of categories that are too problem (or at least too domain) specific. To overcome these problems, Purcell et al.<sup>32</sup> applied three different classifications (They analyzed the same design protocol that we are concerned with here. However, the unit of analysis in their study was not the design move). Two of the classification schemes pertain to what they call “problem domain,” and center around the problem that is being solved: the first classification is called “level of abstraction,” and the second is called “function, structure, and behavior.” As we shall see shortly, we have adopted the first classification in the present study, under a slightly different title. The third classification is of a different kind: it is called “strategy,” and it describes the designer’s activities. We used a similar classification in our analysis.

Purcell et al.<sup>33</sup> developed their coding schemes for the purpose of describing and modeling the design process. Our study is more specific: it looks at a hypothetical correlation between parameters of contents and parameters of structure in front-edge design reasoning. Based on contents alone, we have established critical moves in our protocol, using linkography. For an analysis based on structure, we have developed two coding schemes: one using categories of designer activity, and the other categories pertaining to the level of description of the designed entity. The first scheme is context and problem independent; it is akin to the third classification of Purcell et al. The second scheme is directly adopted from Purcell et al., (where it is called level of abstraction) and is not entirely context and problem-independent.

### 5.1 Coding Scheme 1: Designer activity

Coding Scheme 1 (C.S.1) comprises three major categories: A—*search and development*, B—*reinforcement* and 0—*remarks*. These are divided into subcategories as follows:

- A1** *Proposal* dealing with function or performance
- A2** *Clarification* of functional aspects
- A3** *Analysis* (of function or performance)
- A4** *Explanation* (of function or performance)
- A5** *Assessment* or evaluation (of function or performance)
- B** *Support of / reference* to previously expressed idea(s)
- 0** *Remarks* (agenda; jokes; miscellaneous)

32 Purcell et al., *Analysing Design Activity*.

33 Ibid.

We consider these categories as structural because they are independent of variables pertaining to subject matter, task definition and scope, working style, etc. In our design episode, the functional performance of the entity that is being designed (bicycle carrier) is predominant in the design process, but the categories hold for cases in which other factors might be of greater importance. For example, appearance and aesthetic considerations may be overriding in graphic design assignments. We could argue that, in such cases, it is the *function* of the designed product (poster, say) to attract attention. Therefore, we see the above categories as a generic *structural* coding system for design moves. Table 1 lists the coding of the moves it includes in the column C.S.1. Moves 16, 18 and 27 are examples of coding into category A1; moves 16, 25 and 26 are examples of coding into category A3 (note that double-codes are accepted, as in move 16, which pertains to categories A1 and A3). Category B is assigned to move 20, and move 52 belongs to category 0.

Using these structural categories, we coded the moves in units 32–37 of our protocol.<sup>34</sup> Each move was coded once, sometimes twice (av.=1.06), according to the categories it reflects. Categories A2–A4 were grouped together for the sake of a clearer analysis (a more detailed analysis which was attempted obscured the results, as in the case criticized by Ericsson and Simon). Table 3 presents the results for critical moves (as derived from the linkographic analysis) only.

Table 3  
Category Distribution per CM, Coding Sequence 1

Unit	32		33		34/5		36		37		Total	
Cat.NCM	<8	11>	<0	2>	<10	12>	<10	9>	<7	7>	<35	41>
A1	2	7	0	0	1	2	4	5	1	6	8	20
A2–A4	5	3	0	0	6	5	4	2	4	1	19	11
A5	1	0	0	0	0	1	0	0	1	0	2	1
B	1	5	0	1	2	1	2	1	1	3	6	11
C	0	0	0	1	0	4	1	2	0	0	1	7
Total	9	15	0	2	9	13	11	10	7	10	36	50

Notes:

- 1 <CMs> are counted twice.
- 2 The discrepancy between the total number of nominal CMs and categorized CMs is due to the fact that some moves have a double coding and are, therefore, counted twice.

34 Coding was undertaken by three judges. Two judges reached full agreement, while the third was not in agreement with their results on most counts. Therefore, we discarded his judgment and used the coding of the other two judges.

## 5.2 Coding Scheme 2: Level of description

The categories in Coding Scheme 2 (C.S.2) differentiate between levels at which the designed entity is being considered, by comprehensiveness, as follows:

- I *System level* the entire designed entity and/or its context (bicycle, backpack, and carrier)
- II *Subsystem level* part of the designed entity and/or its context (bicycle or carrier, or backpack)
- III *Details* (joints and the like)
- 0 *Remarks* (agenda; jokes; miscellaneous)

These categories are, at least partially, structural because they are independent of variables pertaining to subject matter (of the designed entity). However, they are not necessarily entirely independent of task definition and scope, design phase, and working style. It is easy to conceive of stretches of time in which designers deal with minute detailing only, and during such phases, we are not likely to find moves that pertain to categories I and II. Likewise, the nature of the task or the phase of designing may call for global considerations only, in which category III may be all but entirely absent. Table 1 lists the coding of the moves it includes in column C.S.2. Moves 16, 25 and 26 are examples of coding into category II; moves 18, 20, 27 and 52 are examples of coding into category III. Move 52 also belongs to category 0.

Using these structural and semi-structural categories, we coded moves in units 32–37 of our protocol (see note 33). Each move was coded once, sometimes twice, according to the categories it reflects (av.=1.04). Table 4 presents the results for critical moves only (as derived from the linkographic analysis).

Table 4  
Category Distribution per CM, Coding Scheme 2

Unit	32		33		34/5		36		37		Total	
Cat. \ CM	<8	11>	<0	2>	<10	12>	<10	9>	<7	7>	<35	41>
I	1	1	0	0	2	4	1	2	0	1	4	8
II	7	9	0	2	4	4	1	1	2	2	14	18
III	2	5	0	0	4	4	8	7	5	5	19	21
0	0	0	0	0	0	0	1	0	0	0	1	0
Total	10	15	0	2	10	12	11	10	7	8	38	47

Notes:

- 1 <CMs> are counted twice.
- 2 The discrepancy between the total number of nominal CMs and categorized CMs is due to the fact that some moves have a double coding and are, therefore, counted twice.

### 5.3 Hypotheses

The categories in C.S.1 and in C.S.2 are perpendicular to one another. Therefore, we can formulate two different sets of hypotheses, each pertaining to one category scheme. The hypotheses are based on the assumption that there is a relationship, indeed a correlation, between properties of structure and properties of contents in design reasoning, and that these properties are reflected in the protocol. Two sets of hypotheses pertaining to critical moves define the correlation that we expect to find:

#### *Hypotheses related to Coding Scheme 1*

Hypothesis 1c.s.1: The coding of <CMs tends to pertain to one or more of the categories A2-A5, B and 0. <CMs do not tend to pertain to category A1.

Hypothesis 2c.s.1: The coding of CMs> tends to pertain to one or more of the categories A1-A4 and 0. CMs> do not tend to pertain to categories A5 and B.

#### *Hypotheses related to Coding Scheme 2*

Hypothesis 1c.s.2: The coding of <CMs tends to pertain to the categories II, III and 0. <CMs do not tend to pertain to category I.

Hypothesis 2c.s.2: The coding of CMs> tends to pertain to category I, II and 0. CMs> do not tend to pertain to category III.

It follows from these hypotheses that forelinking critical moves and backlinking critical moves have equal tendencies to pertain to categories A2-A4 and to category 0 in C.S.1. Likewise, they have equal tendencies to pertain to category II and to category 0 in C.S.2. We propose that the hypotheses related to C.S.1 are stronger than those related to C.S.2 because, as noted above, the categories used in C.S.2 are not entirely context independent.

### 5.4 Method

Each CM was coded for C.S.1 and C.S.2. For each hypothesis, every critical move was marked [+] if it confirmed the hypothesis and [-] if it did not confirm the hypothesis. A <CM> received a double mark. Critical moves with a double coding were marked [+] or [-] if both instances either confirmed or disconfirmed the hypothesis; if one coding confirmed the hypothesis and the other did not, it was marked [+/-]. In a final count, all [+/-] marks were discarded. The final count is given in Table 5 for C.S.1, and in Table 6 for C.S.2. The counting method explains the differences between the figures in Table 3 and Table 5, and between Table 4 and Table 6.

Table 5  
Hypotheses—Coding Scheme 1

	32	33	34/35	36	37	Total
Hyp. 1c.s.1 +	5	0	8	7	6	26
Hyp. 1c.s.1 -	2	0	2	3	1	8
Hyp. 2c.s.1 +	7	1	10	7	4	28
Hyp. 2c.s.1 -	4	1	1	1	1	8

Notes:

- 1 <CMs> are counted twice.
- 2 The discrepancy between the total number of nominal CMs and categorized CMs is due to the fact that some moves have a double coding and are, therefore, counted twice.

Table 6  
Hypotheses—Coding Scheme 2

	32	33	34/35	36	37	Total
Hyp. 1c.s.2 +	7	0	8	9	7	31
Hyp. 1c.s.2 -	1	0	2	1	0	4
Hyp. 2c.s.2 +	6	2	8	2	2	20
Hyp. 2c.s.2 -	1	0	4	6	4	15

Notes:

- 1 <CMs> are counted twice.
- 2 The discrepancy between the total number of nominal CMs and categorized CMs is due to the fact that some moves have a double coding and are, therefore, counted twice.

## 5.5 Results

For each hypothesis, we have to test whether the coding of each CM tends to pertain to specified categories and does not tend to pertain to others. If the overall tendency reaches a percentage above chance level (50%), the hypothesis is confirmed. If it reaches a percentage below chance level, the hypothesis is not confirmed. For each test, we calculate the percentage by dividing the total number of CMs that support the hypothesis and, therefore, have received a [+] mark (as shown in Table 5 and Table 6) by the total number of backlinking or forelinking CMs, as given in Table 3 and Table 4: 35 (note that this is the nominal number of critical moves, and not the number of encoded moves, which is slightly different because of double counting and omissions due to incompatible coding. See notes to Tables 3 to 6). The calculation is repeated for CMs that do not support the hypothesis and, therefore, received a [-] mark. These calculations yield the following results:

*Hypotheses related to Coding Scheme 1 (see Tables 3 and 5)*

### Hypothesis 1c.s.1:

The number of CMs that support the hypothesis: 26

The number of CMs that do not support the hypothesis: 8

The number of <CMs>: 35

Hypothesis supported:  $(26/35) \times 100\% = 74.3$

Hypothesis not supported:  $(8/35) \times 100\% = 22.9\%$

**Hypothesis 2c.s.1:**

The number of CMs that support the hypothesis: 28

The number of CMs that do not support the hypothesis: 8

The number of <CMs: 41

Hypothesis supported:  $(28/41) \times 100\% = 68.3\%$

Hypothesis not supported:  $(8/41) \times 100\% = 19.5\%$

Note: The percentage of supporting and non-supporting CM's for both hypotheses does not add up to 100 percent. The difference is due to the fact that "ambivalent" CMs with the double coding [+] and [-] were discarded in the support-count.

We conclude that hypothesis 1c.s.1 and hypothesis 2c.s.1 are both supported beyond the chance level ( $p < 0.05$ ).

*Hypotheses related to Coding Scheme 2 (see Tables 4 and 6)***Hypothesis 1c.s.2:**

The number of CMs that support the hypothesis: 31

The number of CMs that do not support the hypothesis: 4

The number of <CMs: 35

Hypothesis supported:  $(31/35) \times 100\% = 88.6\%$

Hypothesis not supported:  $(4/35) \times 100\% = 11.4\%$

**Hypothesis 2c.s.2:**

The number of CMs that support the hypothesis: 20

The number of CMs that do not support the hypothesis: 15

The number of <CMs: 41

Hypothesis supported:  $(20/41) \times 100\% = 48.8\%$

Hypothesis not supported:  $(15/41) \times 100\% = 36.6\%$

Note: The percentage of supporting and non-supporting CMs for hypothesis 2 does not add up to 100%. The difference is due to the fact that "ambivalent" CMs with the double coding [+] and [-] were discarded in the support-count.

We conclude that hypothesis 1c.s.2 is supported beyond chance, whereas hypothesis 1c.s.2 is not supported by the data ( $p < 0.05$ ).

If we average the four results that pertain to support of the hypotheses for both Coding Schemes (three that confirm the hypotheses and one that does not), we find that, on the average, our hypotheses are supported at a level of 70% (71.3% for Coding Scheme 1 and 68.7% for Coding Scheme 2).

**6 Discussion: Contents and structure**

The confirmation of most of our hypotheses indicates that in design reasoning, structure, and contents are, indeed, correlated. One might argue that this is a trivial finding, since we obviously expect an evaluative move to deal with previous, not with future

moves; whereas a new proposal has less to do with past moves and would, in all likelihood, be dealt with in subsequent design acts. However, as linkographic studies show, design reasoning is not that simplistic. We should remember that our definition of "move" pertains to a very small design output unit (its average duration is 4.3 seconds) that, on the basis of contents, makes a large number of links. Every critical move generates an average of 10 links, but only part of them are in the "designated" direction (see Table 2). Because "contents" is determined by common sense, intricate backlinks and forelinks may be expected at what may be called a "semantic level": it is sufficient for any common element to be present in two moves to potentially establish a link among them. Therefore, a move's category/link-direction relationship is by no means obvious. In addition, the correlation we have established results from contents and structure analyses that are perpendicular to one another and, therefore, no circular or causal relationships exist between structure and contents as defined here.

It is interesting to inspect the results we have obtained for the hypotheses related to Coding Scheme 2, which were only partially confirmed. C.S.2 deals with the level of description, distinguishing among three levels in a hierarchically descending order (system, subsystem, and details). We have noted that these categories are not entirely independent of task definition, scope, design phase, and working style (see section 5.2). In the bicycle carrier task, for example, each design phase tends to focus primarily on one level of description, from the system level at the beginning of the process, through subsystems, and on to details in the last phase. Our protocol is taken from the middle of the second half of the design process, in which the system level is seldom evoked. Therefore, it is hardly likely for any design moves to pertain to this level of description (category I) in that phase of the process. Hypothesis 2c.s.2, which was not confirmed, anticipates that the coding of CMs would tend to pertain to category I, among others. But since category I is underrepresented in this phase, it is hardly surprising that the hypothesis was not confirmed.

We conclude that, in terms of correlating contents with structure in design reasoning, there is a significant difference between a coding scheme that is "purely" structural and one that is only partially structural. Context-dependency influences correlation values in particular ways that are specific to the task conditions. In our case, only hypothesis 2c.s.2 was effected for the reasons explained above; whereas hypothesis 2c.s.1, with which the context did not interfere, was not effected. For future reference, it is, therefore, important to take special care to avoid context-dependent coding schemes when issues concerning the structure of reasoning are at stake.

It is particularly illuminating to look at the correlation values we obtained: an overall average of 70%, and 71.3% for Coding



Scheme 1 (Coding Scheme 2 is of less interest for reasons we dwell on in the previous paragraphs). These values are compatible with the proportion of links made by critical moves in their designated direction (backwards or forwards) which averages 73.2% (see Table 2). These results suggest that the intricate process of design reasoning is nonlinear, in the sense that, at least as far as significant (critical) design moves are concerned, every step is double speared: it moves forward but also makes sure that it is congruous with what has already been achieved, and it validates what has been done thus far with an eye on ways to proceed from that point. Three quarters of the effort is devoted to the "major" immediate goal, and the remaining one quarter is reserved for the "minor," companion goal. We propose that this pattern represents a cognitive strategy that ensures the efficiency and effectiveness of reasoning in designing: it ensures continuity while also guaranteeing that progress is made, and it serves the need of sustaining a solid and comprehensive design rationale for the entity that is being designed. The success of this strategy hinges on an equilibrated relationship between structure and contents, such as we found to be inherent in design reasoning.

Piaget<sup>35</sup> and other structuralists, who were dedicated to the discovery and explication of structure in thought and matter, were highly aware of the complex relationship between structure, often equated with form in their writings, and contents. Structuralism endeavored to reintegrate content with form, and Piaget's version of this proposed integration theorizes, that in a system, contents and form (structure) are nested hierarchically, with "each element being 'content' relative to some prior element and 'form' for some posterior element."<sup>36</sup> This notion assumes a transformational relationship between structure and contents. We believe that in reasoning, the relationship between the two is not hierarchical; rather, contents and structure concurrently describe the state of a system at any given point and, in effective reasoning, they are apparently extremely well coordinated. We hope to have demonstrated that, at least in design reasoning, this is indeed the case.

### Acknowledgments

We wish to thank Kees Dorst, Nigel Cross and Henri Christiaans, who organized the Delft Protocol Workshop and the design experiment on which it was based. We are likewise indebted to the three anonymous designers who participated in the experiment, and to Xerox PARC and Stanford University's EDRC for their help and support in arranging the design session and in taping the protocols.

The writing of this paper was partially supported by a grant to Gabriela Goldschmidt from the Fund for the Promotion of Research at the Technion.

35 J. Piaget, *Structuralism* (NY: Harper Torchbooks, 1971).

36 *Ibid.*, 29.