Design as Interactions of Problem Framing and Problem Solving

Martin Dzbor and **Zdenek Zdrahal** §

Abstract. This paper introduces a model of framing in design. The model takes into account a reflective nature of designing, and it is based on the interplay between two conceptually distinct knowledge sources – an explicit specification of a problem and a solution to it. The approach is novel in the former investigated aspect that is presented as a semi-formal operation of *framing*, i.e. interpretation of a problem using selected conceptual primitives. We argue that the interpretation of design problems lacks a similar rigorous investigation as problem solving received in both design theory and methodology. Furthermore, two design schemas of frame refinement and problem re-framing are discussed and exemplified.

1. INTRODUCTION

While much effort has been put toward computational models of problem solving in design, there are gaps in the problem interpretation. The existing models understand design as a problem of generating solutions to a problem specification, which is assumed to remain unchanged during the design process. Consequently, they neglect the interpretation of the design problem in terms of an explicit specification. This paper presents design as a two-way interaction of articulating and solving the problem specifications.

A *design task* occurs when an agent decides to change the status of the surrounding world [1]. It is a goal-oriented process that articulates the means for realising the desired changes (e.g. in form of technological artefacts). Usually, design is an ill-structured task [2], to which a solution may not be found until significant effort to understand the 'structure' of the problem has been made. Nonetheless, what does it mean to 'give a problem its structure'? Is it possible to model such a structuring or *framing* using formal language instead of vague terminology of 'intuition' and 'insight' [3]?

The need for a problem interpretation reflects the fact that the designers are rarely given a detailed specification of a problem [4]. A specification of a design problem is built from the initial vague desires, and must be subject to the same evolution as a design solution. We believe that a set of statements characterising a desired state may be proclaimed 'a specification' only at the end of design; i.e. once a designer is satisfied with a proposed artefact, and accepts it as a design solution. An idea of 'co-evolving' design solutions and specifications is not new [5]. However, there are limited formal accounts of this phenomenon in the literature.

The initial incomplete requirements are transformed onto an acceptable specification of a design problem and its solution. Partial solutions influence the requirements, and in turn, the modified requirements refine the solutions, thus revealing the aforementioned co-evolution. Empirically, such a co-evolution is associated with the shaping (*framing*) of a design situation. Schön [4] argues that the practitioners 'know' how to achieve their goals, and shape (*frame*) the design situation to reflect this tacit and experiential knowledge. In this paper, we look at the patterns of '*problem framing*', and develop a conceptual model of framing with two illustrative schemas.

To investigate the relationship between the problem specification and solution development, we conducted 24 experiments with design practitioners. They were solving nontrivial tasks from a domain of controllers for large-scale systems. We illustrate our findings on the session designing a controller for a paper-smoothing plant. For additional details of the experiment settings see also [6, 7].

One of the tasks was to design a layout and control strategy for a plant that takes raw wrinkled paper on the input, and delivers evenly thick, smooth paper at the other end. In a design process, we particularly focused on the designer's reflective behaviour resulting in a problem *re-framing*. The illustrative sketches of a plant shown in Figures 1 to 3 are scanned from the designer's notes. These are the main milestones in the designer's reasoning process:

1. an initial principle for smoothing features a pair of rolling drums with paper passing through a gap between;
2. this layout is enhanced, when a designer suggests dampening the raw paper before entering the rolling drums, and drying it afterwards to achieve acceptable performance
   → *introduction of an additional assumption restricting the scope of solution acceptability* (see Figure 1);

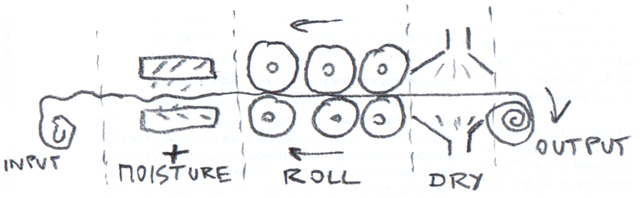

Figure 1. Linear sequence of drums, pre- and post-processing

3. another reflective turn occurs when a designer finds out that smoothing depends on the pressure of the drums, which may damage certain types of paper; an alternative is to reduce the pressure and increase the size of the plant
   → *contradictory requirements are spotted and attended*;
4. layout of the drums is re-engineered (from linear to alternate)
   → *re-interpretation of a concept from the current frame leads to an alternative solution in a new frame* (see Figure 2);

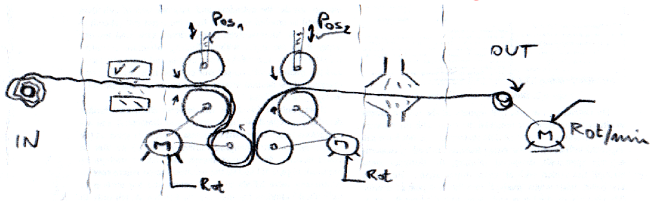

Figure 2. 'Zigzag' sequence → smaller dimensions, better effectiveness

5. the principle of rolling is given up and replaced by 'abrasion' (this is accompanied by a re-design of a solution)
   → *shift in a design perspective (frame), seeing smoothing as an instance of a different physical principle* (see Figure 3);
6. final design solution consists of: (a) pairs of drums to unwind the raw paper and maintain the tension before the output coil; (b) rolling drums 'merged' with dampening mechanism; and (c) from each pair only one drum remains; the drums are positioned in a 'zigzag' manner (see Figure 3, below)

In our opinion, the observed modifications of problem specification cannot be attributed purely to a search for the 'right' solution. The milestones described above feature also a process of *explora-*

---

§ Knowledge Media Institute, Open University, Milton Keynes, MK7 6AA, United Kingdom, Emails: {M.Dzbor, Z.Zdrahal}@open.ac.uk

tion [8] that involves the construction of a design space and interpretation of a design problem. This exploration may show many aspects of intuitive and tacit reasoning but some of its patterns are explicable in terms of evolving conceptual frames and solution acceptability. Let us detail these arguments below.

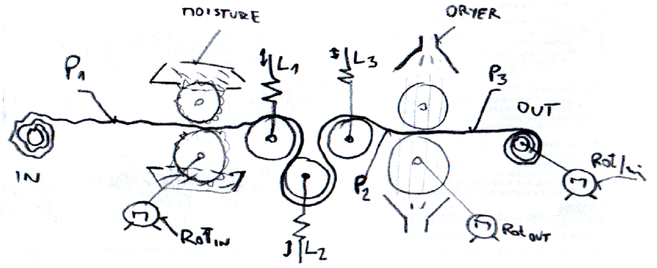

Figure 3. Design solution in a re-interpreted design frame

2. WHAT IS A DESIGN FRAME?

In connection with problem shaping, cognitive science often mentions term 'design perspective' [9]. *A perspective is a point of view, which implies that certain design goals exist, certain bodies of design knowledge are relevant, and certain solution forms are preferred*. The term 'design perspective' is mainly used for expressing a designer's intentions. We may understand a design perspective as a kind of vocabulary of concepts decided upon during problem framing, and used in the problem solving phase of design.

The first major gap in the existing research is an interpretation and a formal clarification of terms 'frame' and 'framing'. The empirical evidence suggest that framing is an important reasoning step that precedes the problem solving and complements it [4, 10, 11]. However, what are the implications of 'framing' on the knowledge level? What is happening with a designer's knowledge during the problem framing? Can this 'framing' be expressed formally? These are some of the challenges we tackle in the further text.

2.1 Essential definitions

Assume a designer is tackling a design problem; let us denote this design problem as $\mathfrak{DP}$. In order to solve design problem $\mathfrak{DP}$, the designer characterises it by an explicit selection of statements from a hypothetical space of potentially applicable problem specifications $\mathcal{S}^*$. Such ***explicit problem specification***, denoted as $\boldsymbol{S} \subseteq \mathcal{S}^*$, provides a context for applicable design methods and domain theories. These methods and theories are tools enabling the designer to satisfy the explicit specification. However, since the problem specification is only an interpretation of $\mathfrak{DP}$, the solution satisfying $\boldsymbol{S}$ must not necessarily be a solution to $\mathfrak{DP}$. This important distinction is discussed more thoroughly later.

We understand problem specification as a set of statements describing the desired states, and expressed in a suitable language; e.g. first-order logic. Formally, we may say that a designer *circumscribes*[1] $\mathfrak{DP}$ by declaring that only the statements from the explicit specification $\boldsymbol{S}$ are needed for interpreting and solving the problem. This circumscriptive step can be expressed by relation (1).

$$\exists S \subseteq \mathcal{S}^*: specifies_{\Phi}\,(S, \mathfrak{DP}) \qquad \textbf{(1)}$$

This assertion can be only made within certain conceptual boundaries – a conceptual design frame. We define design frame $\boldsymbol{\Phi}$ as a pair of two circumscribed knowledge spaces that are constructed on top of the allowed problem specifications $\mathcal{S}$ and the relevant problem conceptualisation $\mathcal{T}$. Thus, 'framing a design problem' means articulating a set of conceptual objects $\mathcal{T}$ that may be used for doing the design, as specifiable by the concepts from $\mathcal{S}$ (relevant problem specifications). Let us use symbols $\mathcal{S}^*$ and $\mathcal{T}^*$ for a formal notation of the circumscribed knowledge spaces constructed on top of both conceptual entities ($\mathcal{S}$ and $\mathcal{T}$).

Space $\mathcal{T}^*$ is 'a closure' constituted by the selected conceptualisation $\mathcal{T}$ and an appropriate domain theory ***DT***. Domain theory ***DT*** is a problem-independent knowledge, possibly applicable to many different problems. For example, physics is a domain theory applicable to a design of elevators as well as spacecraft. However, for different problems, different parts of the domain are used. We say that a generic domain theory ***DT*** is 'instantiated' for a particular conceptual base $\mathcal{T}$, in order to obtain a usable theory for solving the problem. Let us therefore, refer to closure $\mathcal{T}^*$ as a ***problem solving theory***. Similarly, closure $\mathcal{S}^*$ as a hypothetical instantiation of the generic problem specification statements $\mathcal{S}$ in the chosen conceptualisation ($\mathcal{T}$). Finally, we express design frame $\boldsymbol{\Phi}$ formally, as follows: $\boldsymbol{\Phi} = \langle \mathcal{T}, \mathcal{S} \rangle$, and interpret the terms used in its definition:

a) A space of problem conceptualisations $\mathcal{T}$ is an ontology, a vocabulary of basic concepts, for which a designer decides they are available for expressing statements about a particular problem. A conceptual base may include a terminology for the definition of functional and structural objects, as well as problem-specific mappings between the functions and structures, e.g. in form of behaviours [13].
b) A domain theory ***DT*** is a shared ontology, a generic vocabulary defining the background [14], against which any conceptualisation is applied. Domain theory per se is too generic and abstract for problem solving; only its interpretation in a specific conceptualisation yields a usable problem solving theory.
c) A space of relevant problem specifications $\mathcal{S}^*$ complements the problem solving theories $\mathcal{T}^*$. Its principal purpose is to provide a vocabulary for expressing the desires or intentions of a designer in a particular problem [9]. It can be seen as a set of statements that can be formulated about the elements of a particular problem solving theory.

Design frames, as defined above, do not exist 'per se'. They are highly volatile, and are constructed (and re-constructed) on the fly using the information about a particular design problem that is available to a designer. Typically, a designer uses a customer's initial design brief to identify similar design situations, he or she is familiar with. These familiar terms then serve as a seed for articulating an initial explicit problem specification ($\boldsymbol{S}$). The relationship between familiarity and framing is a desirable corollary of our theory, because it relates to empirical observations [4, 9, 10].

After the initial specification the conceptual design continues with a formulation of a minimal sub-set $\boldsymbol{T} \subseteq \mathcal{T}^*$ that satisfies the 'given' problem specification ($\boldsymbol{S} \subseteq \mathcal{S}^*$). Symbol $\boldsymbol{S}$ denotes all such statements that serve to specify the desires about a design problem $\mathfrak{DP}$, to which a designer made a specific and explicit commitment. In other words, a designer tries to shrink the vast space provided by a problem solving theory $\mathcal{T}^*$ into a manageable size that can be manipulated with. This manageable chunk corresponds to the term 'solution model' [15, 16]. Or perhaps, due to its generative nature, it is better to call it a 'problem solving model'.

Formally, a ***problem solving model*** is a minimal sub-set of the problem solving theory that sufficiently satisfies the explicit problem specification. Relation '*satisfies*' is binary, because it associates a problem solving model $\boldsymbol{T}$ with an explicit problem specification $\boldsymbol{S}$, (with the underlying frame $\boldsymbol{\Phi}$ as a circumscribing contextual parameter). Formal definition of a problem solving model is in (2).

$$\exists T \subseteq \mathcal{T}^*: satisfies_{\Phi}\,(T, S) \,\wedge\, (\neg\exists Y \subset T: satisfies_{\Phi}\,(Y, S)) \qquad \textbf{(2)}$$

From an operational point, it is possible to distinguish design requirements $\boldsymbol{R}$ from design constraints $\boldsymbol{C}$, and assert that a problem specification is a union of the two – i.e. $\boldsymbol{S} = \boldsymbol{R} \cup \boldsymbol{C}$ [14]. In this context, requirements are those statements demanding the explicit presence of a particular feature, whereas constraints are conditions

---

[1] This is indeed a circumscription [12] whose purpose is to 'close' designer's understanding of an incomplete, ill-defined design problem.

that must not be explicitly violated by a design solution. More on the conceptual distinction between requirements and constraints can be found in [6, 14]. In this paper, we only present a simplified definition of relation '*satisfies*' in (3) below:

$$\mathit{satisfies}_{\Phi}\,(T, S) \Leftrightarrow \{(S = R \cup C) \Rightarrow T \models R \wedge \neg(T, C \vdash \bot)\} \qquad \textbf{(3)}$$

In (3), the symbols used have their usual meanings [17] – '$\models$' is a semantic entailment, '$\vdash$' is a proof-logical implication, and '$\bot$' is an 'empty' formula, contradiction. Accordingly, theory $\boldsymbol{T}$ is a problem solving model in respect to a given explicit problem specification $\boldsymbol{S}$ and a design frame $\boldsymbol{\Phi}$, if it is complete in respect to the required features ($\forall r \in \boldsymbol{R}$: $\boldsymbol{T} \models r$), and admissible in respect to constraining conditions ($\neg\exists c \in \boldsymbol{C}$: $\boldsymbol{T}, c \vdash \bot$). In other words, a candidate solution must have a potential to deliver all required features without contradicting the constraints.

As already mentioned, the explicit problem specification is only an interpretation of a design problem $\mathfrak{DP}$, which is used in problem solving. It is not the same as problem $\mathfrak{DP}$. Thus, the existence of a problem solving model $\boldsymbol{T}$, for which relation '*satisfies*' holds, is a necessary but not sufficient condition of declaring it a 'design solution'! In addition to satisfying the explicit specification, the discovered problem solving model $\boldsymbol{T}$ must be also '*acceptable*' as a design solution! Unfortunately, a relation of '*acceptability*' cannot be defined explicitly in advance. It is often appreciated subjectively and tacitly, and cannot be expressed in the languages of $\mathcal{S}$ or $\mathcal{T}$.

Nevertheless, it may be defined as a residual category. Formula (4) may help understand the role of the relation of *problem solving model acceptability*'. What does it mean that a relation is residual? We argue that it means the same, as the argument advanced in [18] saying that certain tacit decisions cannot be stripped of their contextual background. It may be difficult to define exact conditions of 'acceptability', but a designer may proclaim a certain problem solving model acceptable or not, when reflecting on it. Tacit reflection on acceptability makes sense only in a particular context, such as frame $\boldsymbol{\Phi}$. Note the change in formula (4) – the contextual parameter for the relation '*acceptable*' is $\mathfrak{DP}$ instead of $\boldsymbol{\Phi}$.

$$\mathit{satisfies}_{\Phi}\,(T, S) \wedge \neg\mathit{acceptable}_{\mathfrak{DP}}\,(T) \Rightarrow \neg\mathit{specifies}_{\Phi}\,(S, \mathfrak{DP}) \qquad \textbf{(4)}$$

We interpret formula (4) so that whenever an otherwise admissible problem solving model is not accepted by a designer as a design solution, it may point to an incorrect (~ incomplete) specification of the actual design problem. The explicit interpretation of a design problem $\mathfrak{DP}$ in terms of statements $\boldsymbol{S}$, does not reflect the real design problem $\mathfrak{DP}$, and it may be desirable to amend it. Such an amendment may feature a refinement of the existing design frame, or a formulation of a new frame (*re-framing*) that are discussed in the remainder of this paper.

2.2 Design as a sequence of conceptual decisions

In the following paragraphs, we propose a recursive model of framing in design using the conceptual entities defined in section 2.1. The model is defined as a sequence of decisions driven by the validity (i.e. returned values) of predicates '*satisfies*' and '*specifies*' (both defined in section 2.1). The sequence is running across several mutually dependent, conceptual levels; in Figure 4 they are labelled '0' to '5'. The model shows the interaction of two distinct knowledge-level actions represented by predicates '*specifies*' and '*satisfies*'. The former action is amending the problem specification, the latter attempts to solve problem interpreted (specified) in this way.

We define also the predicates representing the aforementioned re-interpretations. These definitions intuitively interpret the model as a sequence of *decisions* followed by *actions*. The simplest form of design problem re-interpretation attempts to explicate a statement that is believed to refine the current specification. If such a statement can be articulated within the current frame $\boldsymbol{\Phi}$ (and the conceptual languages $\mathcal{T}$ and $\mathcal{S}$), design continues with a *refined frame* and refined explicit specification as shown in Figure 4, level 3. A refinement within a given conceptual frame is defined in (5).

$$\begin{gathered}\mathit{can\text{-}refine\text{-}spec}_{\Phi}\,(\,S\,) \Leftrightarrow \\ \exists s \in \mathcal{S}^{*}\!: S \subseteq \mathcal{S}^{*} \wedge S' = S \cup \{s\} \wedge \mathit{specifies}_{\Phi}\,(\,S', \mathfrak{DP}\,)\end{gathered} \qquad \textbf{(5)}$$

Slightly more complex decision shown in (6) features an attempt to re-interpret the existing explicit problem specification in a new frame. The problem specification is still refined; however, the new statements are introduced from a new design frame rather than the existing one. Schema (6) differs from one in (5) in the fact that the refinement must be accompanied by a change of the conceptual frame (i.e. the currently used conceptual categories from set $\mathcal{S}$). This technique tries to resolve the 'tacit' non-acceptance of a solution by committing to the terminology borrowed from a new or different design frame.

$$\begin{gathered}\mathit{can\text{-}reframe\text{-}spec}\,(\,S, \Phi\,) \Leftrightarrow \\ \exists \Phi_{\mathrm{NEW}} = \langle \mathcal{T}, \mathcal{S}_{\mathrm{N}} \rangle\!: S' \subseteq \mathcal{S}_{\mathbf{N}}^{*} \wedge \mathit{specifies}_{\Phi\mathrm{NEW}}\,(\,S', \mathfrak{DP}\,)\end{gathered} \qquad \textbf{(6)}$$

The most radical form of problem and frame re-interpretation is defined in (7). In this particular case, the conceptual foundation of the current conceptual frame $\boldsymbol{\Phi}$ (i.e. $\mathcal{T}$ – set of selected conceptual objects) is not consistent with the explicit problem specification $\boldsymbol{S}$. An admissible solution cannot be found in the current problem solving theory $\mathcal{T}^{*}$, because it is inconsistent. To restore the consistency, a new conceptualisation $\mathcal{T}_{\mathbf{N}}$ is chosen. New frame $\boldsymbol{\Phi}_{\mathbf{NEW}}$ is articulated so that the current explicit specification $\boldsymbol{S} \subseteq \mathcal{S}_{\mathbf{N}}^{*}$ is consistent, and an admissible interpretation of a design problem exists (albeit in a different frame, different conceptual vocabulary).

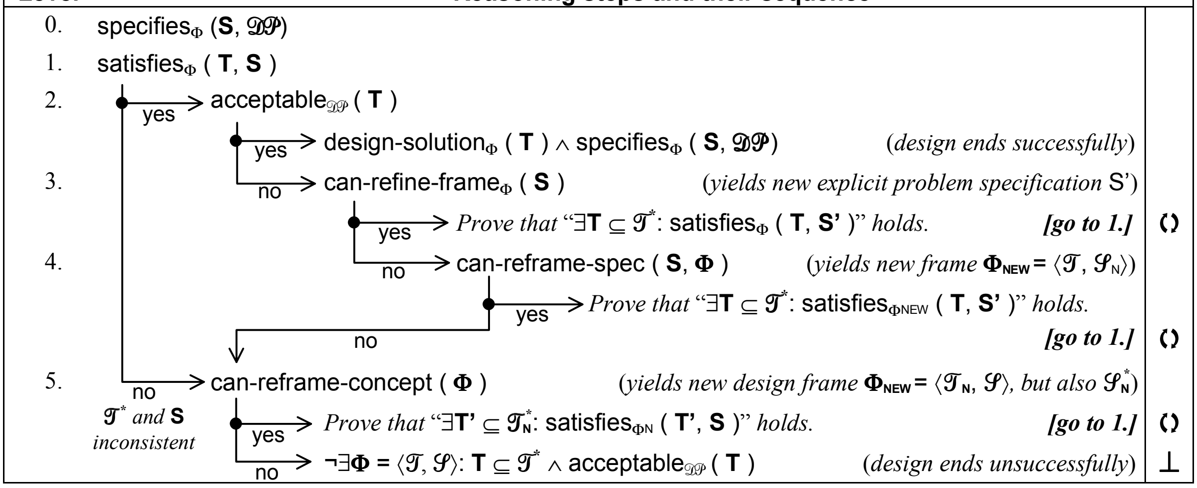

Figure 4. Recursive model of framing in design – a sequence of design decisions on the level of conceptual frames

$$\begin{gathered} \textit{can-reframe-concept}\ (\Phi) \Leftrightarrow \\ \exists \Phi_{\mathrm{NEW}} = \langle \mathcal{T}_{\mathrm{N}}, \mathcal{S} \rangle \colon T' \subseteq \mathcal{T}^{*}_{\mathbf{N}} \wedge S \subseteq \mathcal{S}^{*}_{\mathbf{N}} \wedge \textit{specifies}_{\Phi\mathrm{NEW}}\ (S, \mathcal{DP}) \end{gathered} \tag{7}$$

The recursive model depicted in Figure 4 reflects the empirical observations of the *knowledge-level interactions* that emerged from our experiments. The interactions between complementary knowledge sources are observable in an exchange of information and control between predicates '*specifies*' and '*satisfies*' during a design process. In a construction of the model we made these assumptions:

1) When using terms 'requirements' and 'constraints', we always mean hard, strict demands that *must not be relaxed*.
2) A monotonic extension of a problem specification corresponds to a designer's attempt to '*fine tune*' a problem solving model, to *reduce* the number of derivable alternatives. Problem specification can be refined, only if an admissible problem solving model exists for the current conceptual frame.
3) Sentences "*Try proving that* λ *holds.*" represent *a recursive step* returning to level 1 of the model, and a designer's attempt to address the unresolved issue by amending one or another available knowledge space. It is a kind of 'order' to an agent to *evaluate the predicate* λ *with the new arguments provided.*

Let us describe two reasoning schemas that are explainable directly from the proposed recursive model. Note that these are not problem solving methods. They are proposed as abstract models of certain types of reasoning that may be observed in a conceptual design. Due to limited space, we discuss in detail only two patterns showing frame amendment. First, it is a non-monotonic introduction of new knowledge *refining the frame* in a form of assumptions. The second schema is an example of *conceptual re-framing*.

2.3 Frame refinement & implicit assumptions

Consider the following situation that was observed in the design of a paper-smoothing plant. This section refines milestone 2 from section 1, and in the recursive model is represented as a path leading to the predicate '*can-refine-frame*$_{\Phi}$ *(S)*'. At a certain stage, a designer considered a sequence of rolling drums[2] that applied pressure on the raw paper, thus reducing its thickness, and smoothing its surface. When deriving the consequences of this simple approach, he observed that the effectiveness of both operations depended on the actual pressure and the 'active' surface of drums. The higher the pressure, the greater the expected quality. Nonetheless, paper was a relatively fragile medium with certain limits in respect to tension, and it could tear, if certain limits were exceeded.

The designer thus could not accept a simple sequence of rolling drums as an acceptable solution, despite the completeness and admissibility of the candidate in respect to the explicit specification. The paper was smoothed and thickness was reduced as desired. However, the designer assumed another condition that was never mentioned in the customer's initial brief. In addition to requiring paper smoothing and thickness reduction, he also demanded the paper remained whole (i.e. not torn or otherwise damaged). In a justifying record of this introduction of new knowledge, he maintained that it was "*an obvious condition often not emphasised explicitly*".

When we attend to this apparently straightforward situation, we see that the conceptual base $\mathcal{T}$ for a problem interpretation remained unchanged. The addition of a new assumption monotonically *refined* the explicit specification ($\mathcal{S}^{*} \supseteq \boldsymbol{S'} = \boldsymbol{S} \cup \{s\}$). However, this monotonic refinement had implications on the otherwise non-monotonic problem solving theory $\mathcal{T}^{*}$ with a candidate solution ($\boldsymbol{T}$). The new assumption rendered the 'old' candidate solution $\boldsymbol{T}$ inadmissible; the new condition was obviously violated by the 'old' candidate solution. A new problem solving model $\boldsymbol{T'}$ was found, and it featured pre- and post-processing units (in Figure 1 labelled as 'moisture' and 'dry'). These additional commitments further refined the problem specification, requiring a softened paper, so that a lower pressure was needed, and the danger of tearing eliminated.

What happened in this situation from a knowledge-level point of view, can be seen as an alignment of an explicit conceptual frame with the designer's implicit or tacit expectations. These 'tacit' assumptions and expectations are 'hidden'; however, they may tacitly influence a designer's decision on the solution acceptability. When an admissible candidate solution is found unacceptable, these expectations may become useful. Reflecting on the 'hidden' (perhaps experiential) expectations, the designer may become aware of 'forgotten' features, and s/he explicitly articulates a new statement to address such a feature. With a new statement, the existing problem solving theory may become inconsistent, and may need to be conceptually amended. However, the actual addition of new concepts is a topic covered by a different reasoning schema in section 2.4.

2.4 Conceptual re-framing & contradictory theory

Consider another type of re-interpretation that was observed in the design of paper-smoothing plant, as well as in other experiments. This section refines points 3, 4 from the milestones in section 1, and focuses on the predicate '*can-reframe-concept (Φ)*'. A sequence of rolling drums with pre- and post-processing units in Figure 1 depicts a candidate solution at a certain stage. This solution had no apparent weakness; it complied with the designer's experience with similar problems (e.g. metal sheet rolling). However, when a designer took into account the efficiency and controllability of the smoothing operation, a new issue emerged. As already mentioned in section 2.3, higher pressure or larger active surface of rolling improved a low quality of processed paper. The increases in pressure were tackled earlier, and it was resolved to add the additional processing steps to soften the paper, rather than increase the pressure.

Adding more pairs of rolling drums to the sequence could increase the active surface. Nonetheless, the sequence could not grow forever, because a larger size implied a more difficult control. It was clear that trying to design an assembly with fewer drums was desirable in order to simplify the plant operation. However, fewer drums affected the quality and increased the danger of damaging paper by a higher pressure that was needed. Thus, the designer found himself in a 'magical circle' of contradictory requirements.

He resolved the contradiction by shifting his conceptual foundation. Instead of squeezing or expanding the rolling assembly in 'one dimension' (i.e. linearly laid-out drums), he articulated the concept 'two-dimensional layout' originating in a more cumbersome 'two-dimensional squeezing'. This shift towards a new concept introduced an alternate (zigzag) layout of the drums that featured larger effective surface of rolling. Thus, fewer pairs were needed, and the size- as well as pressure-related constraints could be managed – simultaneously. The new concepts are clearly visible in a re-designed assembly (see Figure 2 in section 1).

A similar reasoning step introducing new concepts to tackle an outstanding conflict was repeated in milestone 5 (see section 1). In this step, the designer made another conceptual *re-framing*. He revisited his initial interpretation of rolling as the principle of paper smoothing. Instead focusing on the pressure application, he became aware that in a zigzag layout one drum was using a much larger surface than the other one in a pair. Hence, he removed a 'redundant' drum from each rolling pair, and replaced the principle of pressing with a principle of abrasion. The components of the plant were similar as before but their conceptual roles were re-interpreted in the new frame, eventually leading to a design depicted in Figure 3.

Unlike in section 2.3, where a problem specification was only monotonically refined, this operation went far deeper. It began with a contradictory problem solving theory $\mathcal{T}^{*}$, in which some constraints were always violated (i.e. $\exists c \in \boldsymbol{C} \colon \mathcal{T}^{*}, c \vdash \perp$). Since none of the violated conditions could be 'retracted', the designer was forced

---

[2] Let us mark this candidate solution by symbol $\boldsymbol{T}$, as defined in section 2.1.

to re-visit the applicable domain theory, as interpreted in conceptual terms $\mathfrak{T}$. Having defined new conceptual primitives (e.g. '2D layout' or 'abrasion'), he actually changed his conceptual vocabulary for interpreting and solving design problem $\mathfrak{DP}$.

A new conceptual base $\mathfrak{T}_N$ triggered articulation of a new conceptual frame Φ_{NEW}, and in the context of new frame, the conflicting constraints 'lost their edge'. The 're-conceptualised' problem solving theory regained its consistency, and design could continue – until other 'hidden' expectations in the next steps invalidated the current perspective. The schema for handling contradictions gives a theoretical, knowledge-level background to an *empirically* observed resolution of physical contradictions in the inventive designs by referring to less-usual conceptual vocabulary [15].

3. DISCUSSION & CONCLUDING REMARKS

The schemas proposed above are defined using knowledge-level concepts and conceptual design frames. They model selected reasoning patterns, which conceptually underpin a designer's evolving understanding of a design problem. It is interesting to investigate, what is going on 'inside' the relation '$satisfies_{\Phi}(T, S)$' or 'inside' the defined re-framing techniques. However, this paper is concerned with a more abstract level of investigation. Although the details of operational models of the conceptual schemas are beyond the scope of this paper, let us discuss a few remarks in that direction.

Why is a logically admissible design questioned? We already mentioned that one reason is the difference between an explicit frame and the implicit and 'tacit' expectations. The adjective 'tacit' deserves more attention because it seems to be closely related to problem framing. In section 2, we defined design frame as an interpretation of an ill-structured problem using a familiar vocabulary of the similar design cases tackled in the past. The 'tacit' and implicit expectations draw on these familiar, past situations. A designer may perceive a similarity between the current and previous cases on different levels of abstraction. Sometimes, the analogy may be too abstract or too complex to articulate it in the explicit terms of a chosen conceptualisation. Simpler analogies may be re-used in tackling the new problem easily; the more abstract ones may remain 'hidden'.

According to formula (4), these past experiences may be hard to articulate explicitly in advance, as standalone analogies. Their essence comes forward in the context of a particular conceptual frame with a particular problem specification and a candidate solution. The examination of consequences of a particular commitment may raise a designer's awareness of inadequacy in the current approach. Such is the origin of the frame refinement described in section 2.3. Past knowledge of pre-heating metal slabs before rolling was contrasted with a lack of any similar operation in an otherwise analogous problem. The designer went to investigate the reasons of preprocessing in the past case, and became aware of material flexibility. Eventually, a notion of improved flexibility was translated into the 'paper' context as a new assumption on the paper fragility, and an 'intuitive' articulation of a new condition became clearer.

Similarly, the conceptual re-framing in section 2.4 may seem confusing and sudden. It surely is sudden and unexplainable within the particular conceptual frame. To explain it we may relate the framing to a '*temporary*' circumscription of an incomplete problem space. By committing to certain explicit conceptual primitives, the context for problem solving is deliberately circumscribed. However, the closure may be re-opened, and the problem circumscribed in a slightly different manner. We believe this is the main difference of the proposed recursive model in comparison to the existing research [1, 8, 12, 13, 16].

The other models typically start with an assumption of a 'given' problem specification, and focus on the solution. In other words, they are concerned with *solving* the problems. In this paper, we made a step towards a more formal conceptual basis of *interpreting* the problems by solving them. In this aspect, the research reported in this paper, extends the empirical work on *reflection* and *evolution* in design [4, 11]. Moreover, the problem interpretation through the interaction between problem framing and problem solving addresses the exploratory nature of design mentioned at the beginning [8]. Design problems are inherently open; by framing they can be 'closed' (circumscribed), in order to address their incompleteness, complexity, and poor structure.

Similar models of design appear in [1, 11, 18]; however, we attempted to present a more formal description of the conceptual phase of design. Although this model is too abstract for the purpose of implementation, it can be seen as a set of guidelines that can serve as a seed of a structured and principled methodology for design support. We hope to elaborate this principled version of the recursive model of framing in more detail, including its operational details and implementation in a real-world setting.

ACKNOWLEDGEMENTS

The authors wish to thank their colleagues from KMi and the anonymous reviewers for their valuable comments. Martin's research has been supported by a joint KMi/Open University scholarship.

REFERENCES

[1] T. Smithers, *et al.*, Design as intelligent behaviour: an AI in design research programme. *AI in Engineering*, **5**(2):pp.78-109 (1990).

[2] H.A. Simon, The structure of ill-structured problems. *Artificial Intelligence*, **4**:pp.181-201 (1973).

[3] R.L. Dominowski and P. Dallob, *Insight and Problem Solving*. In *The Nature of Insight*, R.J. Sternberg & J.E. Davidson (Eds). MIT Press: USA, pp.33-62 (1995).

[4] D.A. Schön, *Reflective Practitioner - How professionals think in action*. USA: Basic Books, Inc., 1983.

[5] S. Nidamarthi, A. Chakrabarti, and T.P. Bligh. *The significance of co-evolving requirements and solutions in the design process*. In *Proc. of 11th ICED*. Finland (1997).

[6] M. Dzbor and Z. Zdrahal. *Towards Logical Framework for Sequential Design*. In *13th ASME Intl. Conf. on Design Theory and Methodology,* Pittsburgh, USA. pp. DETC01/DTM-21710 (2001).

[7] M. Dzbor and Z. Zdrahal. *Towards a Framework for Acquisition of Design Knowledge*. In *27th ASME Design Automation Conference,* Pittsburgh, USA. pp. DETC01/DAC-21049 (2001).

[8] A.K. Goel, Design, Analogy, and Creativity. *IEEE Expert*, **12**(3): pp.62-70, (1997).

[9] K. Nakakoji, T. Sumner, and B. Harstad. *Perspective-based critiquing: helping designers cope with conflicts among design intentions*. In *Proc. of Conference on AI in Design (AID'94)*. pp.449-466 (1994).

[10] G. Fischer. *Domain-Oriented Design Environments*. In *Proc. of 7th Knowledge-Based Software Engineering Conf.*, pp.204-213 (1992).

[11] J.S. Gero, *Conceptual designing as a sequence of situated acts*, In *Artificial Intelligence in Structural Engineering*, I. Smith, Editor. Springer: Berlin, pp.165-177, 1998.

[12] J. McCarthy, Circumscription – A Form of Nonmonotonic Reasoning. *Artificial Intelligence*, **13**:pp.27-39, (1980).

[13] T. Bylander and B. Chandrasekaran. *Understanding Behavior Using Consolidation*. In *Proc. of 9th Intl. Joint Conference on AI (IJCAI'85)*. Los Angeles, California. pp.450-454 (1985).

[14] B. Wielinga, J.M. Akkermans, and A.T. Schreiber. *A Formal Analysis of Parametric Design Problem Solving*. In *Proc. of 9th Knowledge Acquisition for KBS Workshop*. Banff, Canada. pp.37:1-15 (1995).

[15] G.S. Altshuller, *Creativity as an Exact Science*. USA: Gordon & Breach Science Publishers, 1984.

[16] J.S. Gero, Design prototypes: A knowledge representation schema for design. *AI Magazine*, **11**(4):pp.26-36 (1990).

[17] M. Levin, *Mathematical logic for computer scientists*, MAC project technical report TR-131, MIT: Cambridge, Mass. (1974).

[18] S.D.N. Cook and J.S. Brown, Bridging Epistemologies: The Generative Dance Between Organizational Knowledge and Organizational Knowing. *Organization Science*, **10**(4):pp.381-400 (1999).