INTERNATIONAL DESIGN CONFERENCE - DESIGN 2010
Dubrovnik - Croatia, May 17 - 20, 2010.

TOWARD A PRACTICAL GUIDE TO KNOWLEDGE ENGINEERING FOR PARAMETRIC ROUTINE DESIGN

W. O. Schotborgh, F. G. M. Kokkeler, C. A. McMahon and F. J. A. M. van Houten

Keywords: knowledge engineering, design automation, method

1. Introduction

This paper proposes a methodological approach to knowledge acquisition and modelling in support of parametric design. The source of the knowledge is considered to be an expert designer, without experience in knowledge engineering. The acquired knowledge is used to automate the design process. The scope is parametric design with available knowledge: routine variant and adaptive design.

First, the development process of design automation software is briefly described to identify a number of research areas. Several of these are active fields of research, while others remain relatively quiet. The content of this paper focuses on the knowledge acquisition phase, which includes a method to interview an expert to obtain a model of his/her experience-based knowledge.

1.2 Development process of design automation software

A number of concepts, methods and theories from several knowledge domains are required during development of design automation software, Figure 1. The continuous path through the domains, from left to right, indicates the development process of software based on expert knowledge.

The development process begins with a knowledge source. The design processes of the source are first decomposed to reduce complexity, gain overview and determine suitable system boundaries for the software (step 1). After selection of a design process, the relevant knowledge is acquired and made verbally explicit (step 2). Step 2 is the focus of this paper. Next, the knowledge is modelled (step 3) into a format that is subsequently automated by an algorithm (step 4). When developing several different software applications, the concept of generic software development (step 5) is used to reduce the required software engineering effort. Finally, the user interaction is determined to offer the best interaction and highest added value for the end-user (step 6).

A substantial amount of research has been done on ways to automate design problems. Computational Synthesis (CS) is an active field of research that generates and optimizes designs with topological variety, i.e. the element structure of the design is dynamic. A research overview of CS is given by [Antonsson, 2002]. CS offers support for design processes ranging from shape driven (architectural) design to electro-mechanical systems.

Constraint-based approaches are successfully applied in the field of 3D mechanism and linkage system design in [Hicks, 2006]. The model of the problem is stated as equalities and inequalities, and multiple sets of values that represent (possible) solutions are presented to the user. A constraint-based model describes the problem and direct search optimization is implemented to generate solutions.

Optimization algorithms generate solutions toward an optimum, defined by an objective function. A survey of continuous, nonlinear multi-objective optimization methods for engineering problems is given in [Marler, 2004].

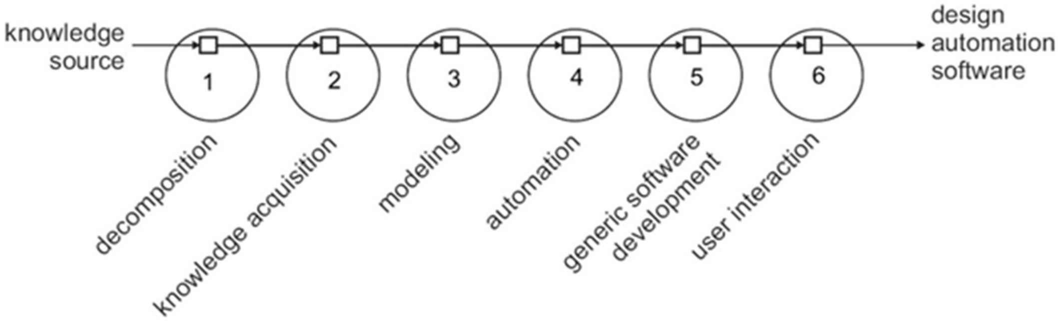


Figure 1. design automation development process

The above mentioned fields of research are positioned in the software development process steps three and four: the development of models and the automatic generation of solutions. These fields are essential for successful development of design automation software in industry. Still, design automation software is not found in industry to the extent that it could be, as described in more detail in [Schotborgh, 2008]. One of the possible bottlenecks is the construction process of a computational model of expert knowledge, i.e. the Knowledge Engineering (KE) process from a human source. The subsequent section addresses the field of Knowledge Engineering in more detail.

1.2 Knowledge engineering

Knowledge Engineering (KE) is the process of constructing a computational model of knowledge: a combination of steps one, two and three in Figure 1. The source of the knowledge is often a human expert and the relevant knowledge is acquired through a process of interviewing, modelling and verification.

In general, KE is seen as an activity that requires understanding of both the computational aspects as well as the design case at hand. [Raphael, 2003] state that the most successful engineering knowledge systems have been created for situations where the engineer-developers were also well acquainted with the subject.

Several KE support tools and methods exist, such as KARL and MOKA. The former, the Knowledge Acquisition and Representation Language KARL [Fensel, 1995] is aimed at knowledge acquisition from the knowledge engineer into a formal language. The Methodology and software tools Oriented to Knowledge based engineering project (MOKA) addresses the issues how to standardize and model design knowledge consistently, once it is gathered. MOKA supports the formalizing process to obtain a model that can be automated [Stokes, 2001].

Perhaps the major difficulty with KE from a human source is the tacit nature of the much expert knowledge and the fact that “design experts are experts in problem solving, not in explaining their solutions” [Fensel, 1995]. The MOKA project found that the largest cost element of building Knowledge Based Engineering systems occurs during the stages of gathering knowledge, extracting the useful parts and representing that knowledge in a structured way [Stokes, 2001].

The tools offered by KARL and MOKA support the knowledge engineer with the construction of a computational model. This project builds on the framework established by KARL and MOKA by proposing tools to support the knowledge gathering phase. This paper describes a method to interview experts and model the knowledge in a way that allows either direct automation, or further modelling into another modelling language. The focus lies on the synthesis phase of design: where design solutions are generated.

First, a theory of a design process and synthesis knowledge is proposed, named PaRC. Second, a KE method is derived to model a design process in terms of PaRC. The method aims to aid the “what to look for and how to get it”- aspect of knowledge elicitation. The method is demonstrated by a case where design knowledge is elicited, modelled and automated to simulate the design process.

2. Model of a design process

This section briefly discusses a model of a design process, its sub-processes and several sets of information.

The left hand side of Figure 2 shows the Function-Behavior-Structure (FBS) model [Gero, 2004]. The solid lines represent the scope of this paper: routine design, which is considered to have structure variations, but it is known in advance what behaviours are taken into account.

This papers' model (Figure 2, right hand side) describes a design process as an iterative loop on a single layer of abstraction. The model is similar to the FBS model, but with definitions that are specifically aimed to model synthesis knowledge parametrically. The input/output information of the processes is explicitly defined.

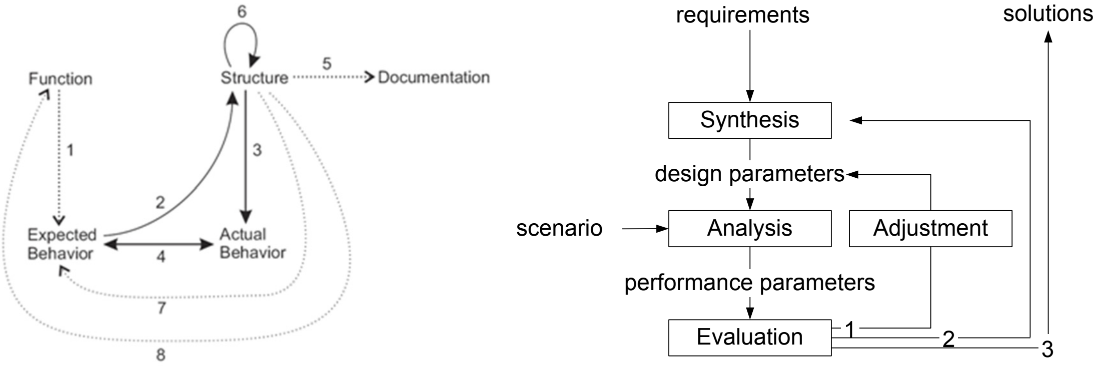


Figure 2. the FBS framework (after [Gero 2004]), design process

First, the sets of information are introduced, after which the sub-processes are discussed.

2.1 Information

This section describes four types of information that are found in design processes as modelled in the Figure 2, right hand side. The difference lies in the way each type is handled during a design process.

The information is described by parameters of different types, such as discrete, continuous, integers, predicates and sets (e.g. a material with a collection of properties).

At the beginning of the design process, a designer is likely to have *requirements.* For a parameterization of routine design, the requirements relate directly to parameter *values*. The goal of the design process is to obtain a design that meets all requirements. Requirements can be fixed numbers, maximum or minimum limits or general optimization goals.

Design

The goal of a design process is to arrive at a description of a “design” that satisfies the requirements. Such a design does not necessarily describe geometry or a physical product, but can also be a layout or sketch. The *design parameters* describe the basic representation of the design artefact or system that is being designed.

The number of design parameters can be smaller or equal than the number of parameters that define any single design proposal. For example, a beam is described by four parameters and a design contains two beams. The design parameters are four, while the design itself contains eight parameters. The design parameters describe the minimum group of parameters that constitute the range of possible designs. Topological degrees of freedom are described by grouping design parameters into elements. Multiple elements construct a design with more than just parametric variations.

Scenario

After a design is generated, it is tested in a certain situation to simulate its usage. The description of such a usage situation is described by the *scenario*: external parameters that cannot be freely changed by the design team. For instance, bridge is being loaded by a number of heavy trucks to check the maximum stress levels. The designer can change the design of the bridge, but not the weight of the trucks. The scenario is typically dictated from outside the design process, such a customer a previous design process.

Performance

After the design and usage situation are both known, the quality of the design in that scenario is determined. The *performance parameters* describe the behaviour that a design exhibits in a scenario. For example, a bridge has a maximum internal stress due to the loads being applied.

This paper focuses on quantitative parametric designs, meaning that the parameters are information entities that receive a value during the design process.

Auxiliary

In addition to the before mentioned independent types of parameters, a fourth type is introduced to express additional (dependent) information such as design intent and temporary construction parameters: *auxiliary* information. Examples are ratios, temporary estimates and the number of parts in an assembly. A model can be expanded with a great number of auxiliary parameters, but it still is the same design.

2.2 Processes

The design process begins with requirements and ends with a solution, through a number of smaller processes as modelled in Figure 2, right hand side.

Synthesis

The first process is where an initial design is generated based on the requirements: the synthesis phase. The name synthesis is used to emphasize its opposition to analysis: synthesis begins with required behaviour and ends with a design, while analysis begins with a design and ends with a specification on behaviour. The information exactly between analysis and synthesis contain the design parameters, which is the core of a design process. The synthesis process is considered a critical activity in the design process. Section 4 proposes a more detailed description of synthesis knowledge.

Analysis

The design parameters have sufficient level of detail to enable analysis. Analysis can only be executed after a design is known, and a scenario is given. Analysis methods can vary from rules of thumb, formulas, spreadsheet calculations to finite element simulations or other dedicated software.

Evaluation

After analysis comes the evaluation. This process compares the analyzed performance with the requirements and decides what to do next. Three options exist:

1. a design seems promising, an adjustment is made to it, after which it re-enters analysis. An automated loop of analysis, evaluation and adjustment results in an optimization process. The optimization method coordinates the design adjustments to steer toward the optimum;
2. a design does not meet the requirements, nor is it expected to. It is abandoned and synthesis is initiated again;
3. the requirements are met and the design is added to the solution list.

Adjustment

Improving a design involves an adjustment process with the goal to better meet the requirements. The difference with synthesis is that both the design and its performance are known.

3. Synthesis knowledge

In a design process, and more specifically the synthesis phase, one could say that the starting point consists of two things: a set of requirements and a set of design parameters that are not yet quantified. The end point is a description of a design that is ready for analysis. It is *knowledge* that enables the designer to go from beginning to end.

The model of synthesis knowledge consists of three parts: the design parameters, a set of knowledge rules and an algorithm to combine these two and perform the act of synthesis.

Background

The model is derived from a cognitive design research's view that "design is most appropriately characterized as a construction of representations. The initial representation is formed by the requirements, and through a series of transformations (e.g. replicate, add, detail, refine, modify and substitute) develops toward its final form" [Visser 2006]. The order in which parts of the representations are modified is described by a strategy.

One such strategy is described as opportunistic, where it depends on the current state of the design and available knowledge to decide on that moment what to do. The particular non-systematic character is attributed to the fact that designers, rather than systematically implementing a structured decomposition strategy, take into consideration the data that they have at the time. This focuses on their knowledge, the state of their design in progress, their representation of this design and the information at their disposal [Visser 2006].

This view from cognitive design research indicates that the synthesis phase starts with the design requirements and elaborates this representation toward the description of a complete design. The role of knowledge during this activity is perhaps not completely clear, but Wittgenstein used the phrase: *"To know means to know how to go on"* (e.g. take the next step from requirements to solution).

An interpretation of Wittgenstein's view on knowledge is that it enables the owner to move from some beginning to some end. The following section proposes a description of synthesis knowledge, based on the view that knowledge enables the owner to take the next step.

3.1 PaRC

The activity of synthesis is the translation of the under-defined input to fully-defined output. Knowledge rules provide methods to determine values for unresolved parameters. These methods can vary from well-informed decisions to more random guesses. First, the modelling entities are introduced, after which the algorithm is described.

Modelling entities

During synthesis, after a value is determined for a parameter, a check is made to verify if the process is moving in the right direction. Again, knowledge provides the means to check if the current set of parameter values is allowed or conflicting. An allowed combination of values describes a valid (partial) design and the next parameter value can be resolved. If the values are conflicting, the design description is invalid. One possibility is to go back some steps and try a different route. The synthesis activity is finished if all parameter values are resolved and without conflict.

So far, three things are needed to describe synthesis for parametric routine design:

1. the information entities: parameters;
2. methods to resolve a parameter value, labelled R-rules;
3. methods to limit the allowed solution space, labelled C-rules.

This description of knowledge is labelled PaRC: Parameter, Resolve and Constrain.

Note: please note the difference between constraints and constrain (without "t"). In general, constraints are all restrictions and relations a solution has to satisfy: both equalities and inequalities. PaRC separates equalities and inequalities by introducing R-rules and C-rules. The R-rules are used to model the equalities. C-rules are the relations that limit, restrict or *constrain* the allowed values of a parameter: inequalities.

Constructing a PaRC model with topological degrees of freedom requires an additional modelling entity: elements. Elements are formed by groups of parameters that belong together, for instance as physical entity or because the design expert addresses them simultaneously. An element is a group of parameters and their R- and C-rules. The topological tree of elements is expanded by X-rules.

An X-rule is executed when sufficient information is known to connect the sub-element(s) to existing parameters. After the expansion, each element will be self-supporting but must "know" its place within the topology.

The synthesis knowledge is organized using the object-oriented paradigm, where the objects are the parameters. Parameters are self sustaining entities, or agents, with their own knowledge rules. It is important to note that these rules are the same for every element, i.e. two compression springs possess the same knowledge.

Knowledge rules consist of three parts:

1. object(s): the parameter(s) or element type(s) to operate upon;
2. conditional set: the set of parameters that is required to have a value before the rule can be executed;
3. action: the explicitly described operation on the object(s).

The general layout of each rule is described in pseudo-code in Table 1.

Table 1. general rule layout

rule	general form
R-rule	if(<condition>) then <parameter value> = <action>
C-rule	if(<condition>) then <parameter value> is valid if: <action>
X-rule	if(<condition>) then 1. create sub-element instances 2. connect to existing design 3. add to parent element

Algorithm

PaRC models are automated by a relatively simple algorithm to simulate synthesis. The algorithm consists of three basic steps:

1. step forward: execute one R-rule or X-rule;
2. check: execute the C-rules;
3. in case of a conflict, reverse one or several steps and try a different route.

The steps are similar to the Role-Limiting Method and Backtracking from constraint satisfaction, but now explicitly defined in terms a parametric design process. A more detailed formulation of PaRC, and the algorithm, is given in [Schotborgh 2008].

4. Knowledge engineering method

This chapter describes a method to construct a PaRC model of synthesis knowledge for a collection of design parameters. The method aims to aid the "what to look for and how to get it"-aspect of knowledge acquisition. In case of a human knowledge source, the knowledge engineer acts as facilitator to make the expert's knowledge explicit.

We assume here that the sub-processes of the design process are clear and we know the design parameters ([Schotborgh 2009] describes the method how to get to this point).

4.1 Parameters

The set of design parameters is the starting point to model the synthesis knowledge as a PaRC model. Parameters are used to reveal the R-, C-rules and finally the elements and X-rules.

Each parameter that is mentioned by the knowledge source is made explicit with a type (design, scenario, performance or auxiliary), a name and a short description, such as stated in Table 2. The Table contains part of the model of the case, discussed in more detail in Section 5.

Table 2. Example parameters

type	name	description
scenario	S_{mat}	sample material
design parameter	T_x, T_y	tube position (x, y)
auxiliary	OC_{prim}	angle primary axis
performance	SB ratio	signal to background ratio

4.2 R-rules

To obtain the R-rules for a parameter (the *object*), specific answers are searched for: the *condition* statement and the *action* procedure. The condition and action statements are acquired by formulating an explicit answer to the following questions:

1. condition: *when* can you calculate a value for this parameter?
2. action: *how* do you calculate the parameter value?

Examples of R-rules are formulas, logic, estimation and random generation between an upper and lower bound. If a mix is mentioned with exceptions or fuzzy decisions, these can be included in the rule. Multiple R-rules are stated if a parameter can be resolved in more than one way.

The condition part tells *when* a rule can be executed: what information must be known? This follows from the content of the action part: all information that is mentioned must be known before it is used.

4.3 C-rules

The C-rules are found by inquiring, for each parameter, if the value is always good and never needs validation. If it is checked somewhere during the synthesis phase, this signals the existence of one or more C-rule. The condition and action statements are acquired by formulating an explicit answer to the following questions:

1. condition: *when* can you check the validity?
2. action: *how* do you check validity?

An example of a C-rule is the check if a value exceeds a certain minimum ratio, such as $a \geq b$. An example of a non-algebraic C-rule for parameter “material” is when certain materials are excluded for specific environments. The condition of the C-rule is true if the environment is known. The action part will exclude all materials from a database that are sensitive to corrosion. The materials have a Boolean to signal if they belong to the allowed set and this Boolean value can be made “false” by the C-rule.

4.4 Elements

Element types are groups of parameters that are addressed simultaneously. The question at this point is: what parameters belong together?

The product that is being designed can be used to determine the element types as physical entities, such as springs, belt drives, x-ray tube and detector. Element types can also be defined according to different functions they perform in a design. For example, an optical chamber (Section 6) has two diaphragm sets that have a different function and are configured differently.

4.5 X-rules

X-rules are revealed as the process that is executed to expand the topology of the design with new elements. Once the element types are identified, the X-rules are formulated to add them. The condition statement of the X-rule must check for the required information to connect the sub-elements to the partial design. Formulation of the X-rules depends on the chosen element types.

4.6 Limitations

The knowledge engineering method is developed for engineering design with existing knowledge, parametric information and quantitative data. Modelling the parameters and their R- and C-rules is relatively strictly prescribed. The model can be modified to the taste of the knowledge engineer, but a first model is made quite quickly.

The scope further includes designs with topological degrees of freedom, such as assemblies and product systems. However, modelling topological entities requires more "personal creativity" of the knowledge engineer. These decisions are less strictly prescribed and therefore lead to more ad hoc modelling solutions. Although it is certainly possible to model designs such as transport networks, the goal of the proposed method is to eliminate the ad hoc approach. The method is most suitable for parametric engineering with fewer topological degrees of freedom.

5. Case

In this section the KE method is applied to an industrial case with tacit, experience based design processes. The design process lies within the domain of optical physics. Only few experts have sufficient (tacit) knowledge to generate designs, with little explicit documentation about their engineering knowledge. Models are obtained from the design, scenario and performance, after which the synthesis knowledge is modelled in PaRC and subsequently automated.

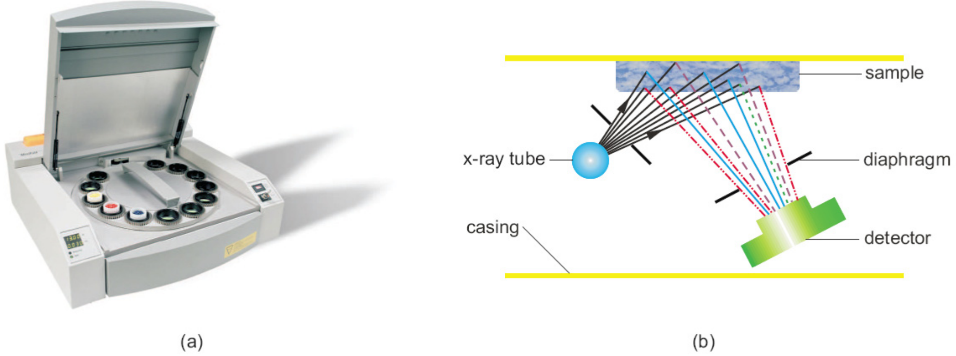
The case concerns the optical chamber design for an x-ray fluorescence instrument, Figure 2(a). This chamber is the heart of the instrument that determines the chemical composition of a sample material. Figure 2(b) shows the construction principle: the x-ray tube radiates the sample material through a diaphragm, and the sample expels characteristic photons into a detector. This in turn reveals the chemical composition of the sample. A solid casing encloses the optical chamber to shield the environment from radiation.

The components should be positioned in such a way that the tube radiates the sample brightly, and the detector "sees" only the radiated section of the sample. Unfortunately, the diaphragms and casing cause unwanted fluorescence and reflections that also enter the detector and negatively influences the measurement quality.

The performance of the optical chamber is specified in terms of its price, size and measurement quality. The measurement quality is mainly determined by two performance parameters: signal-to-background ratio and sensitivity. Signal-to-background ratio describes the relative amount of sample radiation that the detector receives, and the sensitivity represents the absolute amount.

5.1 PaRC model

The design parameters are 22, describing the geometry, position and orientation of the components. The topology is divided into eight elements: tube, detector, sample, casing, diaphragm and two diaphragm sets. The scenario contains 4 parameters: sample material, the tube material and size and detector size.



(a) (b)

Figure 2. XRF spectrometer and design representation

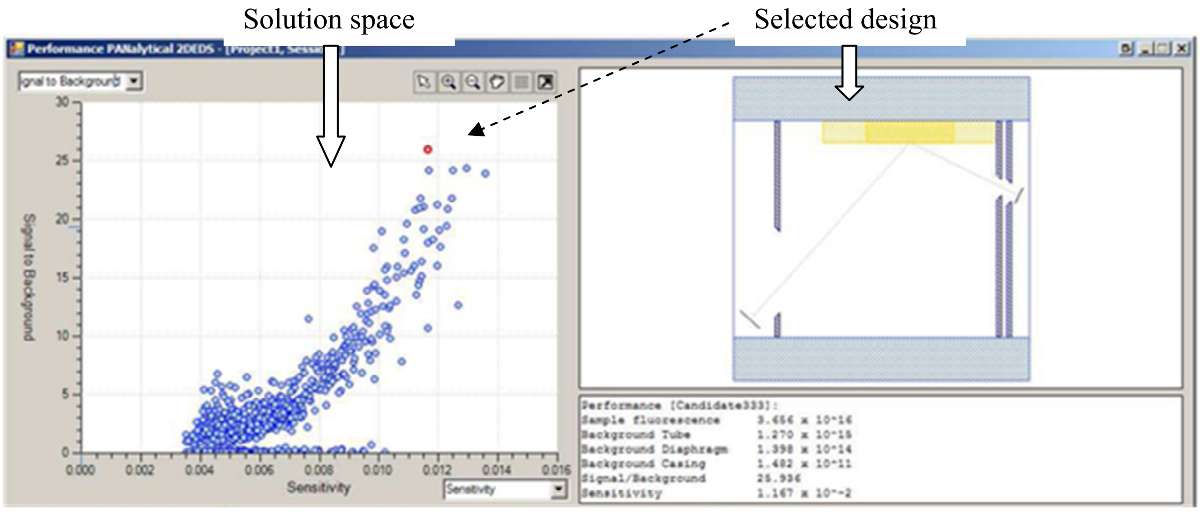
The KE method was executed with the senior expert. All design parameters were mentioned and their R- and C-rules made explicit. Discussing the design parameters with the expert revealed 18 auxiliary parameters to capture the design intent of the x-ray application and several geometric construction parameters.

The R-rules contained a mix of equations, logic reasoning and estimations within bounds set by other parameters. A large group of C-rules was a series of goniometric expressions to prevent the tube and detector from touching the casing. Several others checked material choices, based on maximum energy levels. A total of three X-rules were formulated, together with 19 R-rules and 20 C-rules. Examples of knowledge rules are given in Table 3.

Table 3. Examples of knowledge rules

rule	action part	description
R-rule	$N_P = random\left(N_{P,\min}, N_{P,\max}\right)$	The number of diaphragms at the primary side (tube side) is randomly picked between a minimum and a maximum (set by the user).
R-rule	$D_t = RoundUpTens\left(\frac{\ln(0.1)}{-\bar{\mu} \cdot \bar{\rho}}\right)$	The thickness of a diaphragm is calculated by requesting a factor 10 attenuation of radiation, rounded up to tens of millimetres. The symbols μ and ρ represent derived diaphragm material properties.
X-rule	If the elements sample and tube exist, create N diaphragms and add them to the primary diaphragm set (which is between sample and tube).	

The model is automated to simulate the act of synthesis. Multiple designs are generated and analyzed, as shown in Figure 3. The figure shows a cloud of solutions in the left hand side, plotted in a graph with signal to background ratio versus sensitivity. The graph shows all generated solutions in a solutions space: an overview of what is possible. The right hand side shows the selected design.



```
Performance [Candidate333]:
Sample fluorescence      3.656 x 10^-16
Background Tube          1.270 x 10^-15
Background Diaphragm     1.398 x 10^-14
Background Casing        1.482 x 10^-11
Signal/Background        25.936
Sensitivity              1.167 x 10^-2
```

Figure 3. Design automation tool

5.3 Validation

Validation of the software is done by expert evaluation. Several scenarios and parameter sensitivities are compared to expected values from experience. The quality of the analysis module is validated as proportional to expectations. This means the software can be used to rank solutions according to calculated performances, which is in accordance with the intended use of the tool.

Validation of the generated solutions shows expected characteristics. The expert whose knowledge is implemented expects a significant reduction in design time when these tools would be available in future design processes.

6. Conclusion

A method is presented to model synthesis knowledge from an unfamiliar design process. Tacit expert knowledge is elicited from a (human) source and translated into standardized entities, named PaRC. The entities form a model that can be automated by a backtracking algorithm to simulate the act of initial design generation.

The knowledge engineer is not required to possess the knowledge in order to construct a computational model. The expert designer states the content of the knowledge model in such a way that it is directly useable for automation, based on the identification of parameters, R-rules and C-rules.

Knowledge engineering is becoming a methodological activity that standardizes synthesis knowledge for parametric routine design.

Acknowledgement

The authors gratefully acknowledge the support of the Dutch Innovation Oriented Research Program 'Integrated Product Creation and Realization (IOP-IPCR)' of the Dutch Ministry of Economic Affairs.

References

Antonsson, E.K., Cagan, J., "Formal Engineering Design Synthesis", Cambridge University Press, ISBN 0-521-79247-9, Cambridge UK, 2001

Fensel, D., "The Knowledge Acquisition and Representation Language, KARL", Kluwer Academic Publishers, 1995

Gero, J.S., Kannengiesser, U., "The situated function-behaviour-structure framework", Design Studies, 25(373-391), 2004

Hicks, B.J., Medland, A.J., Mullineux, G., "The representation and handling of constraints for the design, analysis, and optimization of high speed machinery", Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 20: 313-328, Cambridge University Press, 2006

Marler, R.T., Arora, J.S., "Survey of multi-objective optimization methods for engineering", Structural and multidisciplinary optimization, 26: 369-395, 2004

Raphael, B., Smith, I.F.C., "Fundamentals of Computer-Aided Engineering", Wiley, 2003

Schotborgh, W.O., Kokkeler, F.G.M., Tragter, H., van Houten, F.J.A.M., "Why is Design Automation Software not Everywhere?", In Proceedings of the International Conference on Engineering Design 2009, Stanford, 2009

Schotborgh, W.O., Kokkeler, F.G.M., Tragter, H., van Houten, F.J.A.M., "Analysis-Oriented Decomposition of Design Processes", Proceedings of the 10th European Conference of Knowledge Management, Vicenza, Italy, 2009

Schotborgh, W.O., Tragter, H., Kokkeler, F.G.M., M., Bomhoff, van Houten, F.J.A.M., "A Generic Synthesis Algorithm for Well-Defined Parametric Design", proceedings of the 18th CIRP Design Conference 200, 2008

Stokes, M., MOKA consortium, "Managing Engineering Knowledge: MOKA", Professional Engineering Publication, 2001

Studer, S., Benjamins, V.R., "Knowledge Engineering: Principles and methods", Data & Knowledge Engineering 25, pp. 161-197, Elsevier Science, 1998

Visser, W., "Designing as Construction of Representations: A Dynamic Viewpoint in Cognitive Design Research", Human-Computer Interaction, Volume 21, pp. 103-152, Lawrence Erlbaum Associates, Inc., 2006

Wouter O. Schotborgh
University of Twente, Faculty of Engineering Technology,
Department of Design, Production and Management
P.O. Box 217, 7500 AE Enschede, The Netherlands
Telephone: +31 53 489 2266
Telefax: +31 53 489 3631
Email: w.o.schotborgh@utwente.nl