Constraint-Aided Product Design

G. Mullineux, B. Hicks, T. Medland

The importance of supporting the early stages of design is widely accepted. In particular, the development of supportive tools and methods for modelling and analysis of evolving design solutions present a difficult challenge. One reason for this is the need to model both the product design and the design knowledge from which the design is created. There are a number of limitations with many existing techniques and an alternative approach that deals with the design constraints themselves is presented. Dealing directly with the constraints affords a more generalised approach that represents the process by which a product is designed. This enables modelling and reasoning about a product from an often abstract and evolving set of requirements. The constraint methodology is an iterative process where the design requirements are elaborated, the constraint rules altered, design ideas generated and tested as functional structures. The incorporation of direct search techniques to solve the constrained problem enables different solutions to be explored and allows the determination of 'best compromises' for related constraints. A constraint modelling environment is discussed and two example cases are used to demonstrate the potential of a constraint-aided approach for supporting important issues such as the design of product variants and product families.

Keywords: constraint modelling, design knowledge, synthesis, product design, product families.

1 Introduction

It is widely accepted that many of the tasks involved during the early stages of design are critical to the success of a product however defined [1, 2]. A model of the overall design process is shown in Fig. 1. The ability to effectively undertake many of these tasks is largely dependent upon the level of understanding of the design problem and the representation and handling of design knowledge [3, 4]. In fact modelling both the product design and the design knowledge from which it is developed is one of the most important and difficult tasks to support. A number of attempts to address this have been undertaken by academia and industry. These include function-based modelling [5], domain representations [6], ontologies [7] and knowledge based methods [8]. However, the majority of these supportive tools and methods are poor at dealing with generalised problems or problems where the design knowledge is incomplete and continually changes.

One alternative approach is to deal directly with the design constraints themselves. Most design decisions involve considering some form of restriction (real or artificial) on the choices available. Formulating problems in terms of design parameters and these restrictions is a more intuitive approach. Constraint modelling is concerned with formalising and representing design parameters and their relationships as a set of inter-related constraints. The relationships may include mutually conflicting requirements and are generated directly from the design knowledge currently available. As the design proceeds and the design knowledge evolves the constraints set develops. This constraint set represents the process which led to the development of a particular engineering solution.

A constraint-aided process provides a more holistic approach that allows the designer to iterate between the specification, concept and layout stages of the design process. This is shown in the context of the traditional design process in Fig. 1. Constraint rules are derived directly from the specification and used in the evaluation of the proposed functional structure. Failure of this proposed functional description to meet the specified rules results in the re-evaluation of that proposed solution and possibly in a reconsideration of those rules that could not be met. Success leads on to the construction of a preliminary layout in the form of a set of preferred values of a parametric model. Similarly the parametric model itself contains rules that control the limits of its geometry. As a result other failures in rules may occur that need to be resolved by further modifications of the selected specification rules, the chosen concept and the geometric configuration.

Manipulation of all these parameters against the goals set by the rules, also provides information to the designer on those aspects of the design that are highly sensitive to variation and conversely those that have little to no effect upon the solution. This information is invaluable in the process of deriving further or alternative concepts.

This paper presents a constraint-aided approach for supporting the designer during the early stages of product design. A constraint modelling environment based on the notion of constraint satisfaction is proposed as the basis of a design support tool. The approach addresses the issue of modelling and reasoning about the design of products from an abstract set of requirements. It also demonstrates how design knowledge can be incorporated and handled during the early stages of the design of a product. The constraint modelling environment is described and a number of examples are used to illustrate the approach. The examples demonstrate the process of creating rules from the specification, testing ideas as functional structures as well as against the specification, generating preliminary layouts and selecting the best preliminary layout. Furthermore, the examples demonstrate the potential of a constraint-aided approach for supporting important issues such as the design of product variants and product families.

2 Constraint-aided modelling

The constraint based approach aims to represent what is to be achieved rather than how it is to be achieved. These objectives or goals are represented as 'constraint rules'. These represent the relationship between the design parameters, which must be satisfied if the design is to fulfil all of the requirements. These constraints may include performance and physical requirements of the design and also constraints im-

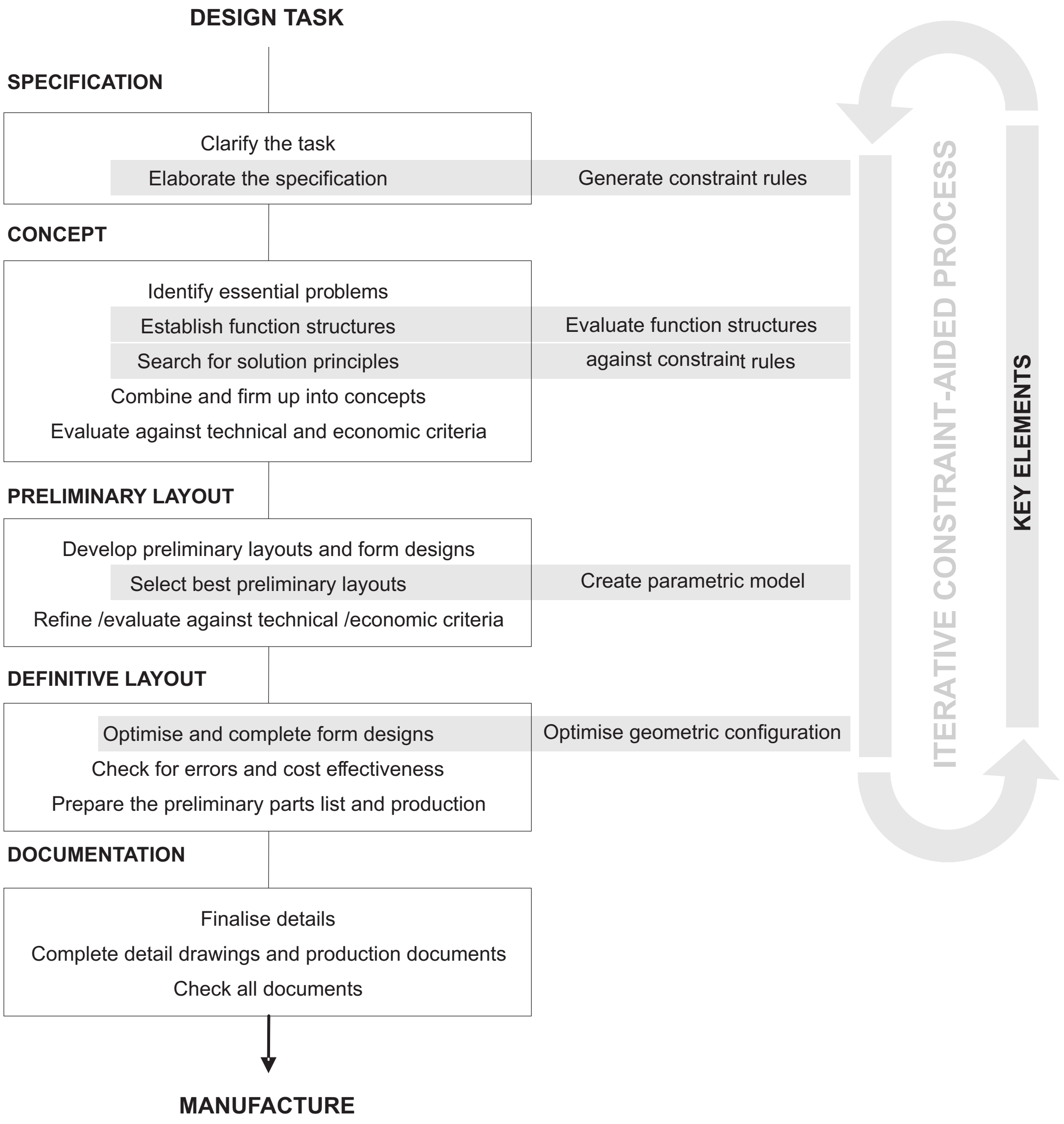

DESIGN TASK
SPECIFICATION
Clarify the task
Elaborate the specification
Generate constraint rules
CONCEPT
Identify essential problems
Establish function structures
Search for solution principles
Evaluate function structures
against constraint rules
Combine and firm up into concepts
Evaluate against technical and economic criteria
PRELIMINARY LAYOUT
Develop preliminary layouts and form designs
Select best preliminary layouts
Create parametric model
Refine /evaluate against technical /economic criteria
DEFINITIVE LAYOUT
Optimise and complete form designs
Optimise geometric configuration
Check for errors and cost effectiveness
Prepare the preliminary parts list and production
DOCUMENTATION
Finalise details
Complete detail drawings and production documents
Check all documents
MANUFACTURE
ITERATIVE CONSTRAINT-AIDED PROCESS
KEY ELEMENTS

Fig. 1: Typical activities in the design process

posed by resources. In this manner the design of the artefact is not process led but goal orientated.

When considering a system it is vary rare that any single element or operation is independent of all the other elements. Consequently, all the goals must be dealt with concurrently and their relationships considered. The aim is to find a solution that satisfies all these imposed constraints as closely as possible. The solution space is the intersection of all the individual constraint fields, as shown in Fig. 2. This intersection is determined computationally by direct search techniques and the manipulation of the design parameters. Using direct search techniques means that convergence on a fully successful solution is obtained if one exists or if not a best compromise is determined. This holistic approach allows the

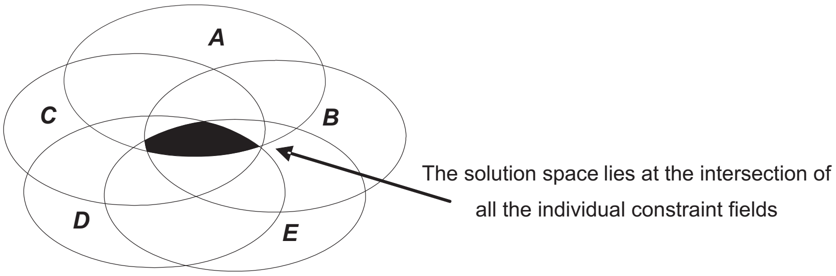

A
B
C
D
E
The solution space lies at the intersection of
all the individual constraint fields

Fig. 2: Constraint rules illustrated as sets

representation of design knowledge, and more importantly, enables this knowledge to be expanded or modified at any time during the process. In this manner, the approach allows changes in both the proposed solution and in the governing constraints of the particular design problem.

In the approach discussed in this paper, no large assumptions about the form of the engineering constraints are made except that it is assumed that the underlying variables are (more or less) continuously varying. Because of this, different forms of constraints between the design parameters can, in principle, be dealt with. The approach implements a general purpose resolution strategy based upon optimisation with the constraints themselves being treated as penalty functions. Although unrefined when compared with specific techniques for dedicated applications, it has proved successful in design software where the forms of the application and constraints are not known a priori.

The language of the constraint modeller has been created to handle design variables of several types including structured types to represent, for example, geometric objects. The language supports user defined functions that are essentially collections of commands which can be invoked when required. Input variables can be passed into a function and the function itself can return a single value or a sequence of values. An important inbuilt function is the "rule" command. Each rule command is associated with a constraint expression between design variables which is zero (as a real number) when true. A non-zero value is a measure of its falseness (error). During resolution the constraint expression for each rule command is evaluated and the sum of the squares of these is found. If this is already zero, then each constraint expression represents a true state. If the sum is non-zero resolution commences. This resolution process involves varying a set of design parameters specified by the user. The sum is now treated as a function of these variables and a numerical technique can be applied to search for values of the parameters which minimise the sum. If a minimum of zero can be found then the constraints are fully satisfied. If not, then the minimum represents some form of best compromise for a set of constraints in which there is conflict. Furthermore, it is possible to identify those constraints that are not satisfied and as a result less important constraints can be relaxed enabling an overall solution to be determined.

As an example, consider the modelling of a four bar linkage. This is shown schematically in Fig. 3. In part (a) of the figure, the two fixed pivot points are declared, and the lines representing the three links are defined, each in a local model space. The model space of the coupler link is "embedded" in the space of the crank, and the spaces for the crank and the driven links are embedded in world space. If transformations are applied to the links in each respective space then partial assembly is achieved. This is shown in part (b) of the figure. If the space of either the crank or the coupler is rotated, the hierarchy of their spaces ensures their ends remain attached.

To complete the assembly, the ends of the coupler and driven link have to be brought together. This cannot be done by model space manipulation alone, as this would break the structure of the model space hierarchy. Instead a constraint rule is applied whose value represents the distance between the ends of the lines [9]. To facilitate this, the user language has a binary function "on" which returns the distance between its two geometric arguments. If *l1* and *l2* are the lines representing the coupler and the driven links, then, in the user language, the constraint rule is expressed as follows

rule(*l1*:*e2* on *l2*:*e1*);

where the colon followed by *e1* or *e2* denotes either the first or second endpoint of the line segment. In order to satisfy this rule, the system is allowed to alter the angle of rotation of the two relevant model spaces. If the rule is applied then the correct assembly is obtained as shown in part (c) of the figure. By rotating the space of the crank link and performing the assembly of the other two links at each stage, a simulation of the motion is achieved, as in part (d) of the figure. If solid objects representing the link are constructed, these can be entered into each model space as shown in part (e).

The following sections discuss in detail two cases which are used to illustrate the supportive capabilities of a con-

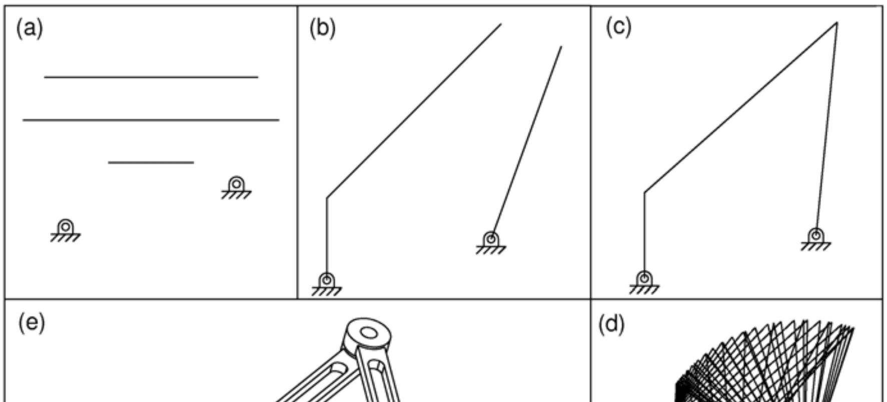


Fig. 3: Modelling a four bar/six bar

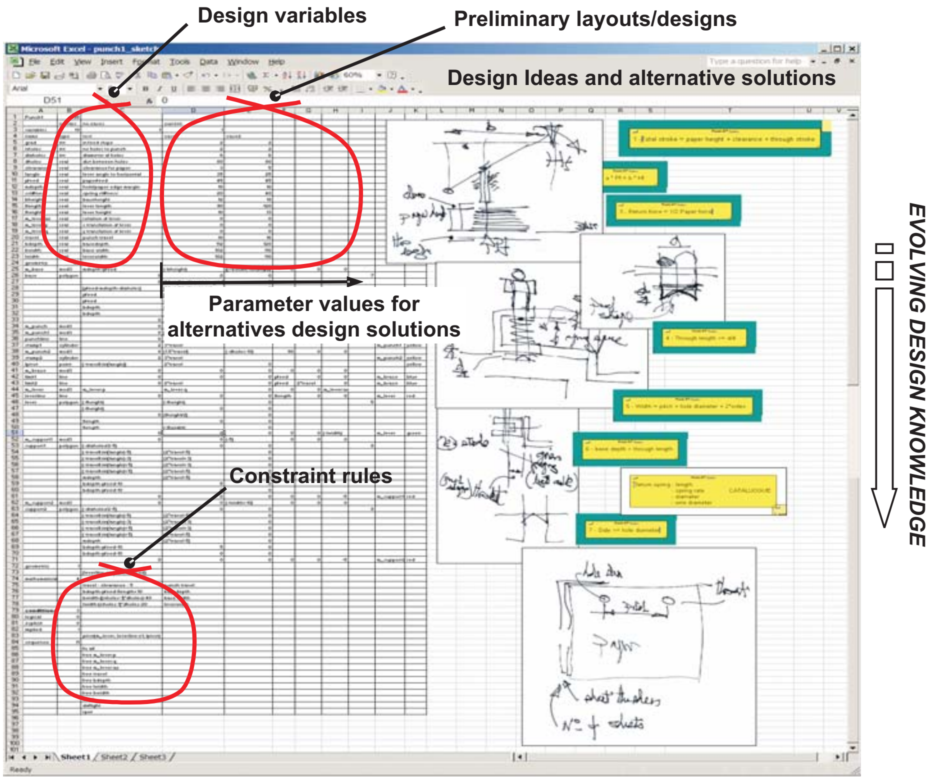

Design variables
Preliminary layouts/designs
Design Ideas and alternative solutions
Parameter values for alternatives design solutions
Constraint rules
EVOLVING DESIGN KNOWLEDGE

Fig. 4: Representing evolving design knowledge for the product development process

straint-aided process for product design. In particular, the examples discuss the ability to generate design ideas and test them as functional structures, embody and evaluate various alternatives or layouts, and identify which performs best. Furthermore, the consideration of product families and product variants during the early stages of design are also discussed.

3 Designing a hole punch

The first example involves the design of a variety of hole punches capable of handling stacks of paper of a variable number of sheets. This example demonstrates in principle the processes of capturing design knowledge, representing and assessing alternatives, and the representation of geometric relations and product assembly.

In the example considered a spreadsheet is used as an intermediary in which illustrations and notes may be added to supplement the constraint rules. The design variables and constraint rules are themselves linked to the constraint modeller by means of the Windows protocol for interprocess communication (Dynamic Data Exchange). These notes and illustrations represent evolving ideas and preliminary layouts. Fig. 4 shows the initial sketches and design notes as well as the associated design variables (geometry) and the conceptual rules (constraint rules). These rules are derived from the product specification and are elaborated as the iterative process of developing constraint rules and exploring ideas is undertaken.

Firstly, a parametric model of the geometry is constructed. The constraints define design requirements or conceptual rules governing the relationships between parts and their assembly operations. The constraint satisfaction problem not only attempts to assemble the product but also to satisfy relations such as ‘base should be greater than handle width’ and ‘overall height must be less than draw depth’. These requirements are expressed as rules that describe the relationships

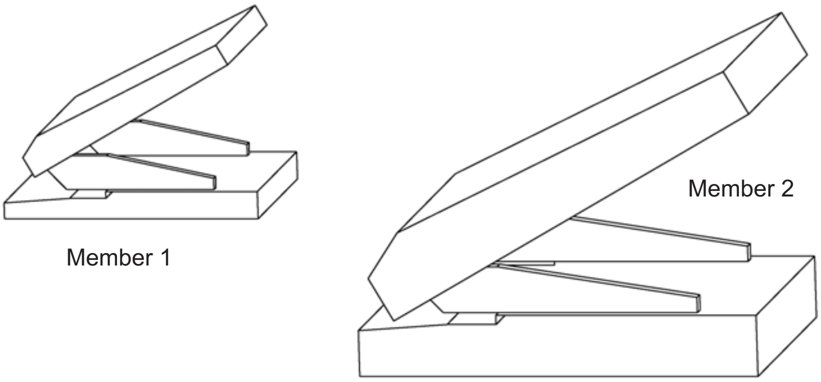


Fig. 5: Product variants determined through constraint satisfaction

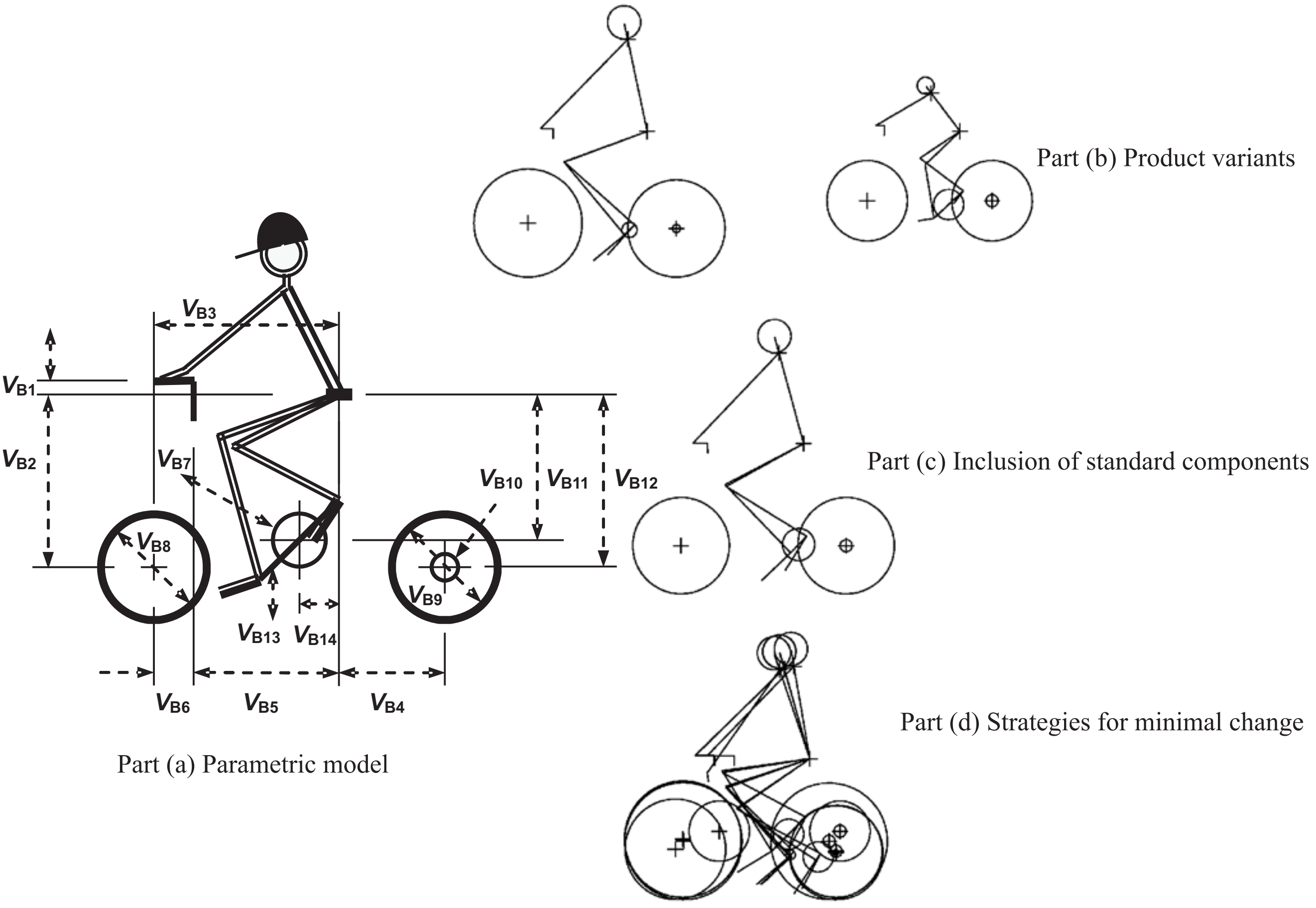
between the design parameters. As the development process proceeds and the design knowledge evolves, the constraint rules alter. By supplementing the constraint set with new and altered rules, a historical record of the outcome or implication of key decisions is constructed. This represents the elaboration of the specification and is particularly useful for design audit or design reuse activities, where a fundamental understanding of the particular solution needs to be obtained rapidly.

Fig. 5 shows two members of the hole punch family as determined by a global solution of the constraints. Design data describing these two specific cases is held within the intermediary spreadsheet shown in Fig. 4. This example illustrates the capability to develop product variants driven by a small number of requirements whilst satisfying or determining best compromises with other essential design criteria. The use of constraints to define assembly operations can be extended to more complex problems such as the assembly of a container involving 11 parts and utilising 30 degrees of freedom [10].

4 A bicycle range

The second example involves the design of a range of bicycles (family of products). In this example, design objectives were firstly collected and a list of design knowledge describing the requirements/specification generated. The design objectives (design requirements) total more than 30 and relate to both the rider and the bicycle. These requirements extend from necessary performance needs and physical restrictions (for example ‘pedal not hitting the ground’ and ‘foot not going through front wheel’) to styling considerations that limit the range and sizes of the wheels. This design knowledge is transformed into a series of constraint rules that relate to a parametric model of the bicycle and the rider. For example, the ‘reach the pedal at the bottom of the stroke’ requires that the total length of the rider is greater than the distance from the saddle to pedal position at bottom dead centre. The corresponding constraint rule is expressed in terms of the design variables for the rider and for the bicycle. Fig. 6 shows a parametric model of the bicycle and its associated design variables.

For the case considered, the objective is to develop a product family. To achieve this, a global solution is sought for the constrained problem for various sizes of rider. Part (b) of Fig. 6 shows two variants for different sizes of rider. In these solutions everything has been varied in order to satisfy the constraint set. However, in practice it is desirable to use a number of standard components. These can be included by either fixing the values of certain design parameter sor by restricting their range of permissible values to within a bounded interval or even a list of values, such as sprocket sizes. In part (c) of Fig. 6 a solution obtained using standard components for the wheels and the sprockets is shown.



Part (b) Product variants

Part (c) Inclusion of standard components

Part (d) Strategies for minimal change

Part (a) Parametric model

Fig. 6: A parametric model of a bicycle and design solutions determined through constraint satisfaction

When developing a product range or family it is often desirable to explore strategies for minimal change. This is made possible by investigating the sensitivity of the design solution to changes in the design variables [11]. Dominant or critical design variables may then be altered to best achieve the desired effect with the minimal change to the solution. A number of strategies to achieve the desired changes in design objectives are shown in part (d) of Fig. 6. Each solution is determined by altering only a proportion of total design variables.

5 Conclusions

The need to model both the product design and the design knowledge from which the product is created has been discussed. There are limitations with many current techniques and an alternative approach that deals with the design constraints themselves has been presented. Dealing directly with the design constraints allows a more generalised approach that represents the process by which a product is designed. Furthermore, dealing with the constraints allows the modelling and reasoning about a product from an often abstract and evolving set of requirements, which is particularly important during the early stages of design. The constraint process is an iterative one where design requirements are elaborated, constraint rules altered, design ideas generated and then tested as functional structures.

The constraint-aided process provides for the representation and handling of design knowledge through the creation of constraint rules. The process of forming constraint rules acts to elicit and formalise design knowledge. The constraint rules not only describe relationships between design parameters but also assembly operations and desired performance capabilities. Direct search techniques are applied to satisfy the set of constraint rules fully or to determine a best compromise. In particular, the constraint-aided approach supports the exploration, evaluation and assessment of alternative design solutions. Design alternatives can be generated by altering the constraint set or the relative weighting between constraint rules, whilst a potential solution can be embodied by specifying desired performance characteristics or system outputs as a set of constraints rules. Finally, design ideas can be assessed by testing them against the set of constraint rules. The example cases presented, also demonstrate the potential of a constraint-aided approach for supporting important issues such as the design of product families and the inclusion of standard components at the early stages of product design.

Acknowledgement

The ideas reported in this paper have emerged from work supported by a number of grants funded by Engineering and Physical Sciences Research Council (EPSRC), Department of Trade and Industry and Department for Environment Food and Rural Affairs (DEFRA), involving a large number of industrial collaborators. In particular, current research is being undertaken as part of the EPSRC Innovative Manufacturing Research Centre at the University of Bath (reference GR/R67507/01). The authors gratefully express their thanks for the advice and support of all concerned.

References

[1] Pahl, G., Beitz, W.: *Engineering Design: A Systematic Approach*. Second Edition, Springer-Verlag, 1996, ISBN 3-540-19917-9.

[2] Pugh, S. : *Total Design*. Addison-Wesley Publishing Company, New York, 1991, ISBN 0-201-41639-5.

[3] Hubka, V., Ernst, Eder W.: *Design Science: Introduction to the Needs, Scope and Organization of Engineering Design Knowledge*, Springer-Verlag, 1996, ISBN 3-540-19997-7.

[4] Miles, J., Moore, C.: *Practical Knowledge-Based Systems in Conceptual Design*. Springer Verlag, 1994, ASIN 0387198237.

[5] Rosenman, M. A., Gero, J. S.: "Collaborative CAD Modelling in Multidisciplinary Design Domains." In Maher, M. L., Gero, J. S., Sudweeks, F. (eds.), *Preprints Formal Aspects of Collaborative Computer-Aided Design*. Key Centre of Design Computing, University of Sydney, Sydney, 1997, p. 387–403.

[6] Chakrabarti, A., Blessing, L.: "Representing Functionality in Design." *Artificial Intelligence for Engineering Design and Manufacture*, Vol. **10** (1996), No. 4, p. 251–253.

[7] Gerhenson, J. K., Stauffer, L.: "A Taxonomy for Design Requirements from Corporate Customers." *Journal of Research in Engineering Design*, Vol. **11** (1999), No. 2, p. 103–115.

[8] Hayes-Roth, F., Jacobstein, N.: "The State of Knowledge-Based Systems." *Communications of the ACM*, Vol. **37** (1994), No. 3, p. 27–39.

[9] Anantha, R., Kramer, G. A., Crawford, R. H.: "Assembly Modelling by Geometric Constraint Satisfaction." *Computer-Aided Design*, Vol. **28** (1996), No. 9, p. 707–722.

[10] Mullineux, G.: "Constraint Resolution Using Optimisation Techniques." *Computer & Graphics*, Vol. **25** (2001), p. 483–492.

[11] Apps, B., Mullineux, G.: "Sensitivity Analysis within a Constraint Modelling System." International Conference in Engineering Design, Proc. 11th ICED, Riitahuhta, A., ed., August, 1997, Tampere University of Technology, Vol. **3**, 1997, p. 309–312, ISBN 951-722-788-4.

Dr. Glen Mullineux

Dr. Ben Hicks
e-mail: b.j.hicks@bath.ac.uk

Prof. Tony Medland

Innovative Manufacturing Research Centre
Department of Mechanical Engineering
University of Bath
Bath, BA2 7AY, United Kingdom