

DESIGN CONTEXT KNOWLEDGE BASED PROACTIVE SUPPORT FOR COMPONENT DESIGN

Xiu-Tian Yan¹, Fayyaz Rehman¹ and Jonathan C. Borg²

¹*CAD Centre, Department of Design, Manufacture and Engineering Management
University of Strathclyde, 75 Montrose Street, Glasgow G1 1XJ, UK.*

²*Department of Manufacturing Engineering, University of Malta, Msida MSD 06, MALTA*

Abstract: The research work described here establishes a new design support approach to product life-oriented design. The approach defines a broad concept of design context information. Having completed the function design, the approach supports the product design through background reasoning using the design context information. This reasoning mechanism can facilitate a designer to explore rapidly different design alternatives generated through function to its realisation (means) mapping. “Unfit” means can be identified using the design context information. The approach provides further support in making suggestions by relaxation as to how to avoid a solution candidate being identified as an “unfit” life-oriented design solution. The architecture and its implementation of a prototype KICAD system entitled FORESEE II is detailed in the paper. The concept of the approach and an illustrative use of the FORESEE II system through a case study are also described.

Key words: Knowledge intensive engineering, proactive life-cycle design, design context knowledge, and design reuse

1. INTRODUCTION

A new product is typically developed through a number of stages, namely: market research, design, prototyping and testing, production process planning, manufacturing, assembly, inspection and sale (French 1985; Pahl and Beitz 1996; Pugh 1991; Swift 1987). Various product realisation tasks have been traditionally arranged to take place as a sequence of activities carried out by different groups of engineers. This consequently leads to a

2.1 Artefact solution's knowledge modelling

By studying existing products using the reverse-engineering approach, mechanical artefacts can be viewed as systems that can be decomposed into systems of a finer resolution (Hubka and Eder 1988). Observing the artefact structure from a constructional viewpoint (Andreasen and Hanse 1996) gives rise to what is termed a product breakdown structure (PBS), typically represented as a compositional ('part of') model. Typical PBS can be a product/system, consisting of some sub-assemblies. Each of these sub-assemblies is typically composed of some basic components. As shown in Figure 1 - a typical sheet metal product, a component can be made of some component-building primitives. In this view, an artefact consists of a number of elements at different levels, termed in this paper product design elements (PDEs) these being:

- *component building PDEs*: component design elements that constitute a component e.g. for a computer power unit's casing, component elements include the blank, slot, hole, material, snap-fits and rib features;
- *component PDEs*: a single material product component produced without any assembly operations; e.g. the base & strip of the power unit are two separate PDEs;
- *sub-assembly PDEs*: i.e. elements consisting of more than one product components. An example is base & support strip assembly of power unit. This sub-assembly is regarded as PDE at sub-assembly level.

It can be seen from the above discussion that Product Design Elements can therefore be generalised and modularised from past design solutions and collected to form a library of well proven and used design solution building primitives. Once these PDEs are formalised into the reusable library, they become an important source of solution ideas to any given new design problem. As these are well-proven partial design solutions, they can be used with confidence to provide satisfactory partial solutions. The overall solution to a given design problem can thereafter be derived by combining these partial solutions for the design problem. In this paper sheet metal products are considered and relevant PDEs are used to illustrate the approach for product solution modelling.

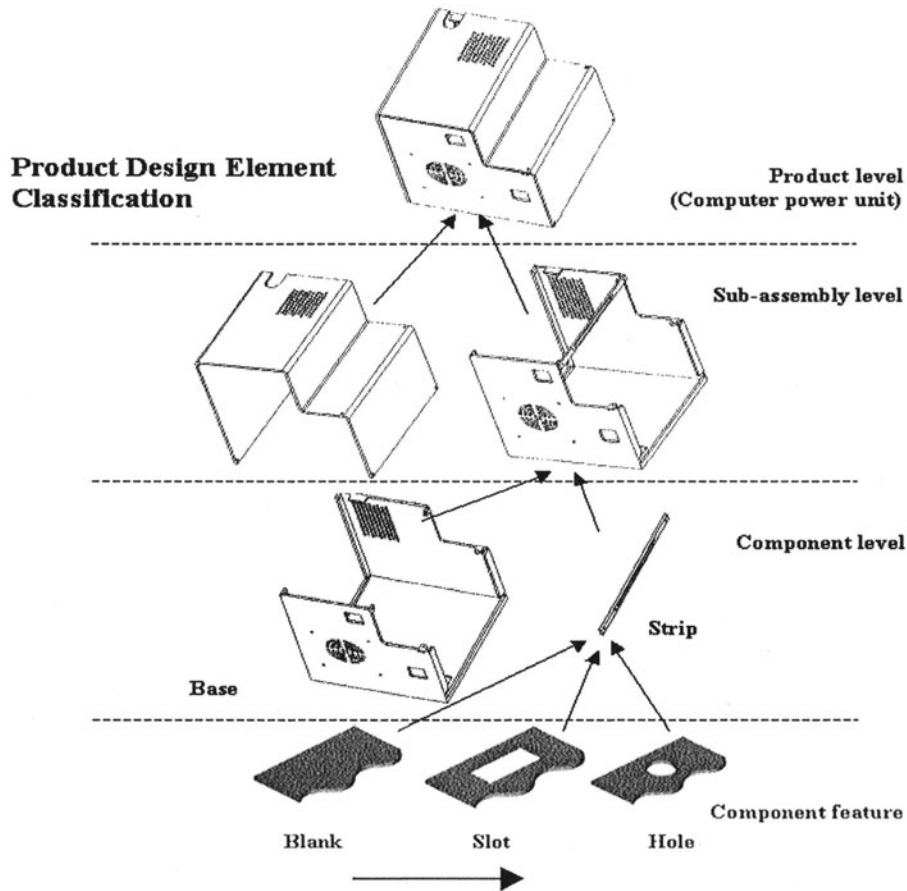


Figure 1. Product composed of PDEs at different levels

This research adopts the above design re-use methodology through modularization of PDEs, and PDE-oriented modelling approach based on the object-oriented programming methodology to handle vast design solution information. The effective combination of these two methods results in a unique methodology to tackle and make effective use of vast amount of design information during the product development. An effective design support is enabled through this Product Design Element based knowledge intensive approach.

2.2 Product-life phase solution's knowledge modelling

A manufacturing system, or more generally any other systems used during a product life-phase, e.g. assembly robots, recycle machines and so forth, can be considered as artefacts, although they tend to be more complex than a normal product. The design of such a system is the same as that of a product artefact. The ranges of information required to define such a system are normally much broader and more complex. Hence more data are required

to describe them. Much research work has been done in the detailed modelling of these life-phase systems and representation of their functions (Janis and Bartholomew 1993; Pham and Ji 1999; Yan et al. 1994). These approaches define methods of modelling one type of these systems in a great depth. Complex information models are required to model the behaviour of these systems. However, the implementation of these models may not interact with a product directly, hence these systems don't support product engineering design directly. As this research focuses on the support of product life-cycle design, information regarding the key characteristics of these systems, rather than modelling the whole system in details, suffice the needs to support product life design. Such a model with reasonable modelling accuracy still is of a great degree of complexity, even only the key characteristics of a life system are considered,. Consequently, an information intensive model is also required for such a system.

Following the same principles of modularization and design re-use, Life-Cycle Process Elements (LCPEs) can be generalised and collected to form a useful life-phase system library. These LCPEs can include CNC machine tool, forming machines, robots for automated assembly, material/part separation machines for product disposal and so forth (see Figure 2). As the emphasis of this paper is on the manufacture/assembly of the product life phases, these *Manufacturing/Assembly Tool Elements* (MATEs) are used to illustrate the philosophy of the life knowledge intensive modelling approach derived from this research.

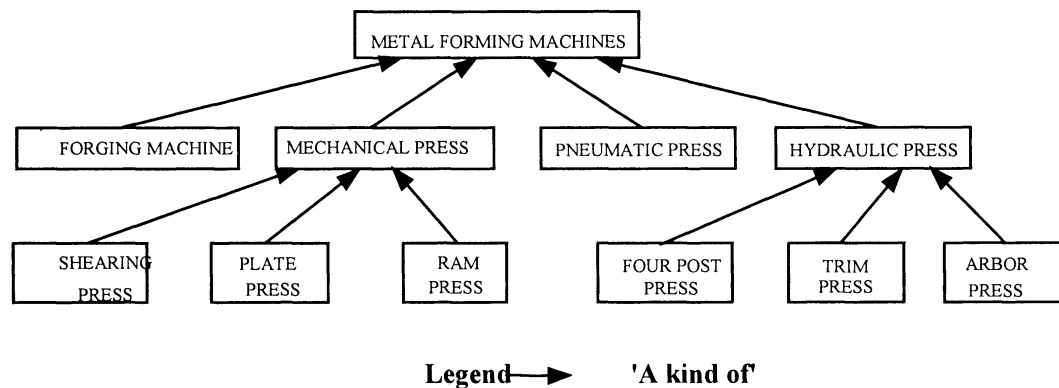


Figure 2. An Example of Manufacturing LCPEs

3. FUNCTION REASONING AND MAPPING

3.1 Function decomposition

Based on the establishment of the above product solution and its life-phase modelling, it is possible to consider the process of design problem solving. This research adopts the design method by Pugh (1991) and assumes that a design process starts with market research to formalise a product design specification (PDS) document. From this document, it is possible and desirable to describe and concentrate on the functional requirements – the key aspects of product engineering design, and decompose the overall high-level function into small and implementable sub-functions. This is due to the fact that rarely one can find a single solution, which can achieve a specified high-level function in engineering design. A well-decomposed function hierarchical structure represents a good understanding of customers' requirements for a product. This is particularly important to function oriented design as such a structure represents the results of the function reasoning process and this is also the basis for the following embodiment design.

The functional requirements are often decomposed to a level where it is possible to identify potential means or solutions. For example, in sheet metal component design, one of the desired function requirements could be *Provide-Assembly* and the potential means from the function means mapping library could be *Lancing*. For sheet metal component design, means of achieving a function are more likely in the form of geometric features. In another word, Product Design Elements (PDEs) at component building level can most likely be possible means of achieving a desired function requirement. For the decomposed function structure, this research proposes the following PDE based function mapping to identify the suitable means to realise a chosen function.

3.2 PDE based design through function-PDE mapping

In addition to function decomposition, function design also refers to the process of generating a design solution from product function point of view, using available well-understood function-PDEs relationships to identify suitable means/mechanisms in the form of PDEs to realise a chosen function. PDEs are the information carriers that allow the mapping between function requirements and physical solutions of a product. They are the basis for embodiment phase, detailed part design phase and later life phases of product. This research derived a dictionary of well-proved PDEs associated

with its enabling function(s). Through this association, the function-means mapping algorithm can be written to identify suitable PDEs for a chosen implementable function. The function design process can be facilitated by the PDE based knowledge intensive approach. All the above have been implemented in a FORESEE system to support such a function reasoning process.

3.2.1 PDE based function reasoning

A PDE is a reusable design information unit (element) representing a potential solution means for a function requirement. Of relevance to this definition and looking from the viewpoint of component construction, a more commonly used term *feature* is considered to be an information element defining a region of interest within a product. The feature description contains the relevant properties including the values and the relations of properties of a product. Furthermore, a feature is described by properties out of several different classes of properties, thus relating these (classes of) properties to one another (Case et al. 1994; Brunetti and Golob 2000). It is argued in this approach that a feature is one type of PDEs at component building level. PDEs are hence used as the key to function-oriented mapping.

Figure 3 shows the process of PDE based function mapping, where a [*Provide-Semi-Permanent-Assembly*] function can be realised by four possible means or PDEs at an assembly level. This is derived from the mapping search algorithm defined for the mapping process. Similarly when a small function can be realised by a component building PDE, it appears as a potential candidate. [*Assembly_slot*] PDE is such an example in which a slot solution is nominated by the mapping algorithm for designer's consideration. The potential solutions generated during PDE based function reasoning need to be evaluated using some design context (background) information based reasoning mechanism in order support the designer in decision making of selection of suitable means. The next section explains how context based reasoning can highlight unsuitable function means to provide proactive support to designer.

4. CONTEXT BASED REASONING TO SUPPORT FUNCTION DESIGN AND VALIDATION

During the process of mapping PDEs to functions, the key objective is to generate as much as possible different alternatives. These alternatives/options can help designers to rationalise their function

requirements through exploration, relaxation, addition or elimination of functions specified. Unnecessary functions can be eliminated; contradicting function requirements can be identified and relaxed. Missing functions can be added to fulfil the technical requirements. For each small-decomposed function, function-PDE mapping can usually result in several alternative PDEs for a designer to consider. Additionally, for an extensive function requirement definition for a product, the alternative PDEs generated through the mapping can be significant, and sometimes the number could be exponentially increasing. In a paper based design approach, designers tend to ignore many alternatives and focus on the ones with some previous experience. This may result in less optimal design solution produced. If a designer wishes to try a new solution, a trial-error approach is normally adopted, as no previous knowledge can be used for the new solution. With the above scenarios, this can be a demanding task if each of these PDEs is fully manually evaluated. Designers as human beings can only handle up to seven chunk of information and they are limited by this capability of handling a large amount of information. In addition, the deadline for a solution design can be quite tight. To support effectively designers in these scenarios, this research proposes and developed partially a reasoning mechanism using design context knowledge. It aims to use important and relevant context information to provide proactive and intelligent design support through background reasoning. Unsuitable design solutions will be highlighted through this context-based reasoning when each of PDEs is explored. Due to an extensive representation of the product solution knowledge and its life-phase solution knowledge as discussed in section 2, using their associated attributes developed in this research, it is possible to use design context information for background reasoning to support the function reasoning.

A design context is defined as the related background information of a design problem under consideration. It provides vague, possibly incomplete and discontinuous, boundary definition information for design solutions. At the same time, it also provides useful hints, constraints to the solution space of the design problem. A good understanding of design context information is essential to successful design. More specifically, design context information can include:

- available information of life-cycle processes which will be used to realise and dematerialise the product;
- information of users of the intended solution products, e.g. age, gender, product preference etc.;
- materials selected for the solution product, more specifically the properties of the chosen material

With the above context information available, the context information reasoning mechanism aims to detect any 'unfit' PDE from the initial mapped PDEs. The reasoning process using design context information involves firstly searching (see Figure 3). The initial function requirement to **Provide-Assembly** has been matched with four possible means to implement this requirement. Searching the context information of solution material selection reveals that aluminium material has been selected for the solution, and the joining part has plastic material type. This activates a piece of knowledge that Soldering means cannot be used for the function, as aluminium components cannot be soldered with plastic component. Timely prompting this context information of material assist designers to eliminate these infeasible options.

4.1 Manufacturing Assembly Consequences (MACs) based function reasoning

Due to the space limit, this paper focuses on the life-cycle context knowledge and uses it to illustrate the approach to the research. Of particular interest to this research is the important group of life-cycle design context information. Life cycle context or more specifically in this paper Manufacture/assembly context information is the definition of life-phase (Manufacture/assembly) systems constraints to product definitions. This group of context information is classified in this research in the library of manufacture/assembly consequence (MACs). These are important source of knowledge about the constraints that manufacturing/assembly systems impose on design decisions of a solution product. Designers are often unaware of these limitations and as design decisions become more related to other factors, it is very difficult, if possible, for designers to foresee these potential decision consequences. This research does not intend to exhaust all MACs for all scenarios, however, it intends at this stage to generate sufficient useful and important MACs to facilitate function reasoning. Through the use of these MACs, it is possible to remind designers proactively the potential consequences of their decisions. These can be very useful insights to the downstream implication of a design decision. As part of the context information reasoning mechanism, MACs can be used in function reasoning. Designers can be proactively supported with timely prompts about the potential downstream implications of a design decision at early stage. For example, committing Hole-Fastener as the selected PDE to realise the function to **Provide-Assembly** for a sheet metal component, triggers of a piece of MACs that violates design for assembly principle as this decision results in more parts for the design. Through the context information reasoning, specifically MAC based reasoning, suitable PDEs for

a function requirement can be reduced. In this example, designers are left to explore the remaining two alternatives.

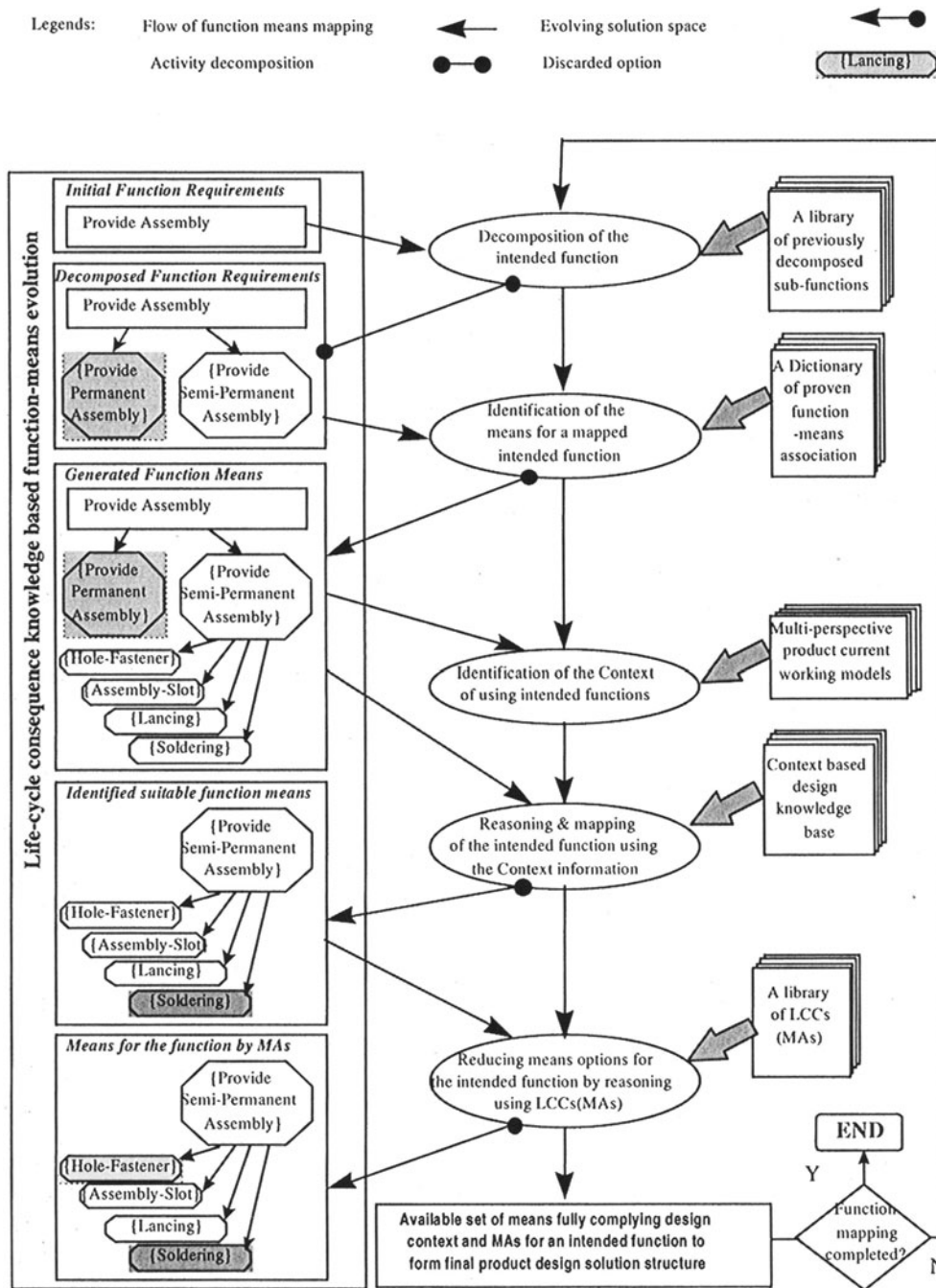


Figure 3. Function to PDEs mapping and reasoning process

It can be seen that using design context information in the background of design process helps designers to process vast amount of potentially related design information and prompts useful insights when they are available

through reasoning. Reasoning using MACs can further assist designers to concentrate on exploring design alternatives and generate more innovative design solutions. The PDE based function-means evolution methodology explained in section 3 & 4 using design context and MACs based reasoning mechanisms forms the basis of developing a framework for implementation of a prototype system.

5. FRAMEWORK FOR IMPLEMENTATION OF APPROACH

To illustrate the effectiveness of the approach, a second *Knowledge-Intensive-CAD* prototype has been implemented recently as FORESEE-II (Yan et al. 2001) for the sheet metal component domain for this on-going research programme. The *integrated* 'product' and 'life-phase' system design approach framework was firstly implemented as 'FORESEE' for the thermoplastic components (Borg et al. 1998).

5.1 FORESEE-II System architecture and implementation

Based on the early discussion, a '*Knowledge of Life-Cycle Consequences*' (*KLCC*) *system architecture* to sheet metal component 'Design Synthesis for Manufacture/Assembly' has been developed (see Figure 4). The extended KICAD system FORESEE-II comprises a knowledge base, working memory and inference engine. The Knowledge base contains detailed representation of reusable synthesis PDEs and MATES features. *Working knowledge* stores the resultant information about the component model and its life-phase model derived from a concurrent synthesis. The *Inference engine* is the domain knowledge independent reasoning mechanism, containing rules/knowledge to reason with the information stored in the knowledge base. The Knowledge base further consists of Inference knowledge, containing design consequence knowledge, and Reusable Product Design Elements Library. This concept is realised in FORESEE-II for manufacturing phase of sheet metal component. The process design involves the selection of tool and machine for the part to be manufactured. Based on the understanding of Manufacturing/Assembly consequences (MACs), it is possible to generate basic Machine & Tooling features from Form features of sheet metal component thus realising the

concept of concurrent product and process design of sheet metal components.

A set of tools has also been designed to facilitate the communication between a user and the Knowledge Base. These include: a Component Model Viewer to visualise the co-evolving model of component, a Form Feature Editor to add or modify form features, a Consequence Browser to see the consequence that would occur during product development caused by design decisions, an Alternative Design Solution Viewer in order to avoid bad consequence and a Tooling/Machine Parameter Viewer to see the design parameters required to manufacture a form feature.

5.2 FORESEE-II Prototype System Implementation

The architecture has been implemented using MS Visual C++ version 6 on Windows NT and an open GL libraries based system called Open CASCADE. The new prototype system has been coded to demonstrate the scaling-up of the FORESEE system.

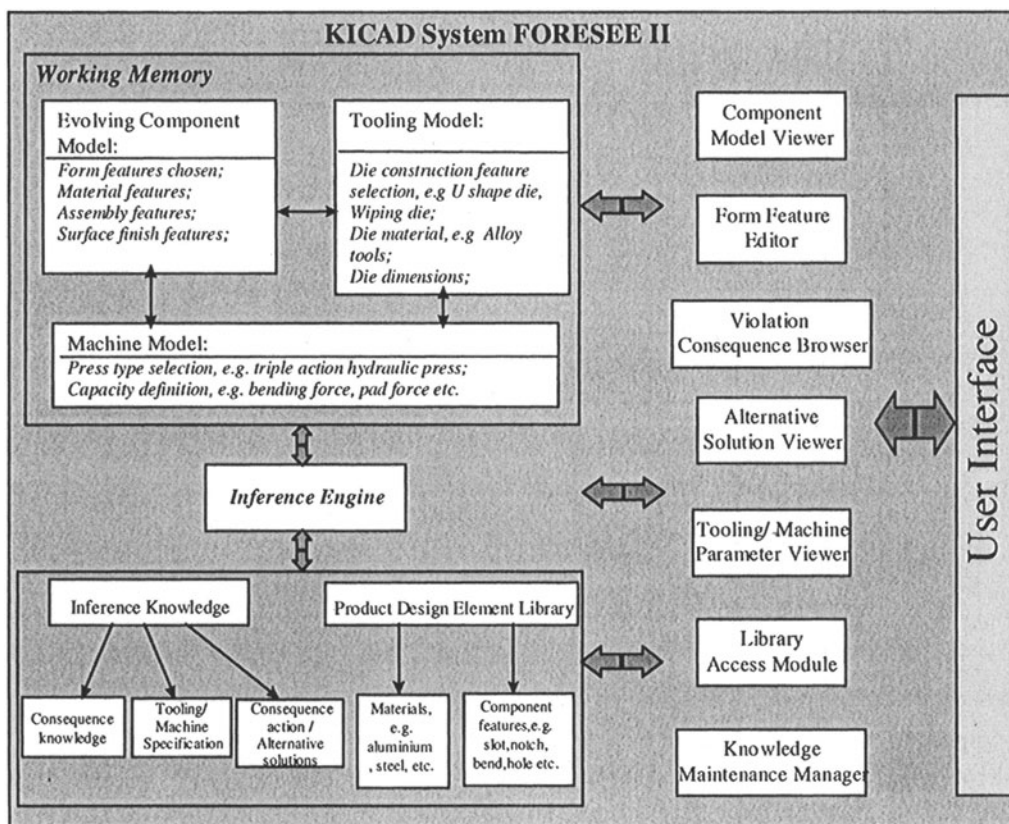


Figure 4. FORESEE-II System Architecture

The prototype has been tested by demonstrating case studies to various researches of engineering design and in the process of further development and refinement. Next section provides an illustration of the use of the system through captured screen images of a case study.

6. CASE STUDY OF SHEET METAL DESIGN

This case study demonstrates the capabilities of FORESEE-II in designing a sheet metal component, using context MACs knowledge. Based on the result of function reasoning by a designer, a *bend* and a *hole* PDEs are required for the component to be assembled onto another base component. Having added an *Unequal Length Bend* to a flat sheet metal as shown in Figure 5(b), successfully, with the help of context reasoning mechanism implemented in FORESEE II in a similar manner as shown below, the following illustrates the scenario of adding the hole. The component properties as well as PDE properties can be defined through the *PDE and Component-Base Editors*. When the hole PDE is added to the component via PDE Editor (Figure 5(a)), FORESEE-II checks in the background of the decision using MAC knowledge. Assume the distance between the centre of the hole to the Bend is specified by the designer to be 10.725 mm as in Figure 5(b). Using MAC knowledge, the FORESEE-II detects that the decision violates a design-for-manufacture rule:- the minimum Hole-to-Bend distance must be greater than $2.5 * \text{thickness} + \text{Bending_Radius}$. The inference engine then uses the above knowledge to deduce a violation message “*INSUFFICIENT BEND TO HOLE DISTANCE*” displayed in the *Violation /Consequence Browser* as shown in Figure 5(c). The inference engine further helps the designer to understand the situation by providing detailed explanations of the violation as shown in Figure 5(d). The proactive design support devised by the system takes a step further in helping the designer to rectify this violation by suggesting several alternative solutions as shown in Figure 5(e). Assume the designer is to change the position of the hole further away from the Bend, the new committed distance of 15.725 mm is then acceptable in Figure 5(f). With the acceptable dimensions specified for the sheet metal component, the FORESEE II system calls its reasoning mechanism to reason in the background for the size of tooling to form the PDE feature. Applying this information, a designer can start to create tooling design model with the help of the Tooling Editor. Similarly, designer can also start to build basic function requirement properties for a press machine.

MAC knowledge in the form of rules is used to help designers to build tooling and machine models. Examples of the MAC knowledge include:

- If *Surface Finish* of component (feature) is very high then *Die Construction* will be *Fine Blanking Die*.
- If ‘Tooling’ selected is ‘Fine Blanking Die’ then Press Type would be *Triple Action Hydraulic Press*.

As different types of punch and die sets (tooling) are used to form corresponding PDEs in sheet metal manufacturing, basic geometry- related dimensions/parameters of tooling used for Die Construction can be derived from PDE (*Form*) features. The interplay of tooling design and component design often results in useful insights to a design. Having defined partial tooling features, a designer can select a Machine for concurrent component and MA design. The machine selected should be capable of producing the required PDE *Form* feature (hole, slot, notch etc.) on the component. Machines used in sheet metal working operation are presses of different nature. Basic design features/parameters of machines (presses) can be generated from ‘Tooling’ and ‘Form’ features. The geometry based reasoning and MAC rules can be used to produce ‘Tooling’ feature from ‘Form’ feature and ‘Machine’ feature from ‘Tooling’ features. This type of consequence awareness at the early stage of design can help the designer to *foresee* the problems that are likely to occur at the later stages of product development. This has been shown in the above case that manufacturability of the component is not guaranteed using current bending machine to produce the *bend* feature specified by the designer. The extended FORESEE system not only detects inappropriate design decisions using the consequence knowledge, but also suggest possible remedies to avoid undesired consequence by generating automatically alternative design solutions/decisions. This has taken a step further to proactively support a designer to explore design alternatives for an optimal solution.

7. CONCLUSION

This paper presented an integrated and interactive ‘product and ‘life-phase system’ *based* approach for supporting ‘concurrent design’. Through the introduction of three-dimensional integrated product total life design modelling solution space, the information intensive nature in concurrent product and its life-systems design is established. This provides the basis of extending the design support to function mapping and reasoning using context information and LCCs. The evaluation of the approach through the implementation of the FORESEE & FORESEE-II prototypes, disclose that the approach contributes a step towards the development of knowledge intensive *CAD* tools that can pro-actively support designers in generating ‘life-oriented’ product solutions. As established, the approach:

- Demonstrates that ‘knowledge intensive’ CAD tools provide suitable means to retain, process and explicitly reuse captured LCC knowledge for guiding designers in searching for life oriented solutions from information intensive multi-dimensional solution space;

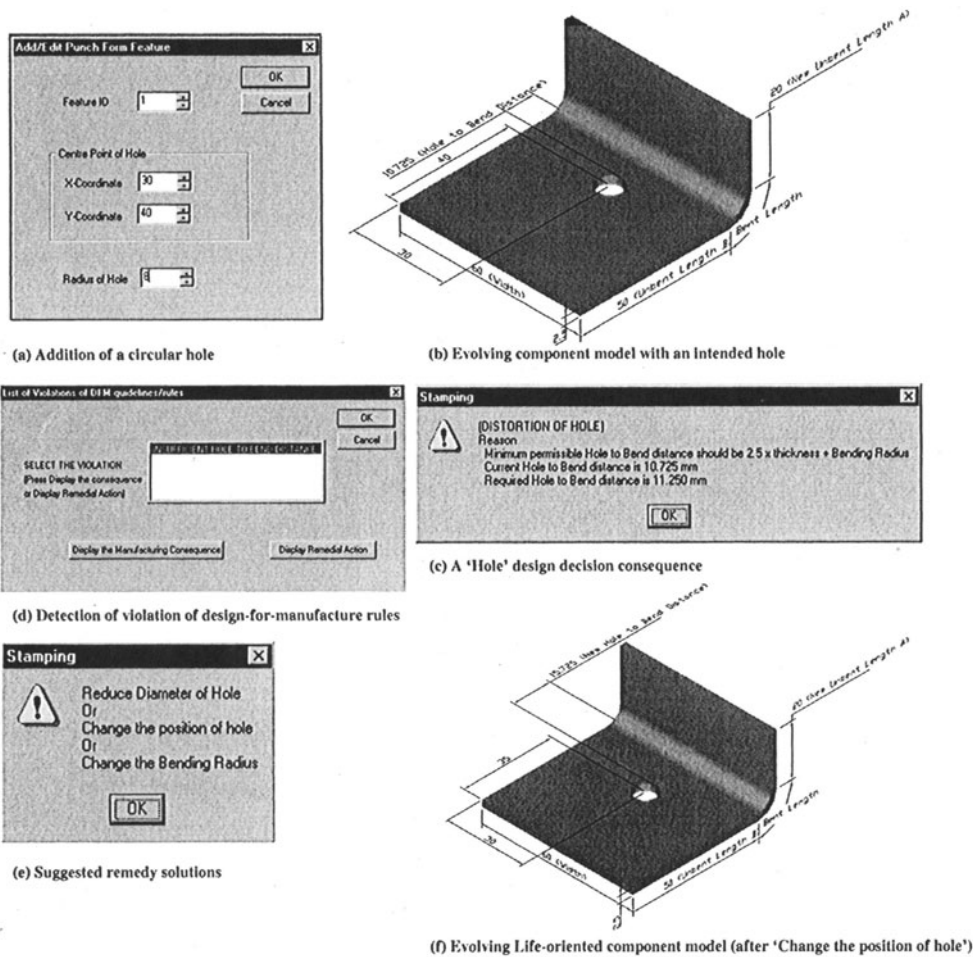


Figure 5. An Illustration of use of FORESEE II

- Reflects that the general system requirements on which the two FORESEE prototypes are based, contribute to the development of *life-oriented design* tools;
- through the experience of extending FORESEE to FORESEE-II, provides a sound foundation for life-oriented product and life-phases systems modeling. Successful extension of FORESEE architecture for injection molding domain into FORESEE-II architecture for sheet metal domain proves the suitability and scaling-up into other applications domains.

As discussed, further work is however required to this approach to broaden its scope of support from component building level design stage of a product to assembly level stage.

REFERENCES

- Andreasen, M. M., Hanse, C.T., "The structuring of product and product programs", 2nd WDK workshop on product structuring, Delft University, The Netherlands, 1996.
- Borg, J.C., Yan, X.T., & Juster, N.P., "A KICAD tool for pro-active exploration support to 'Design synthesis for Multi-X'", In Knowledge-Intensive Computer -Aided Design, Kluwer Academic Publishers, Boston, USA, 1998.
- Brunetti, G., Golob, B., "A feature based approach towards an integrated product model including conceptual design information", Computer Aided Design, (2000), Vol. 32, pp 877-887.
- Case, K., Gao, J. X. and Gindy, N. N. Z., The implementation of a feature based component representation of CAD/CAM integration. *Proc. IMechE, Part B: Journal of Engineering Manufacture*, 208, 71-80 (1994).
- French, M., "Conceptual design for engineers", 2nd edition, London, Design Council, London, 1985.
- Hubka, V. & Eder, E., "Theory of Technical Systems: A total concept theory for engineering design", Springer Verlag, Berlin/Heiderberg, 1988.
- Janis P. Terpenney and Bartholomew O. Nnaji, " Feature Based Design Evaluation for Machine/Tool selection for sheet metal", Proceedings of 2nd Industrial Engineering Research Conference, pp 26-30 (1993).
- Pahl, G., & Beitz, W., "Engineering Design- A systematic approach", 2nd edition, Spring Verlag, London, 1996.
- Pham, D. T., & Ji, C. "A concurrent design system for machine parts", Proceedings of IMechE, Part B: Journal of Engineering Manufacture, 213, pp. 841-847, 1999.
- Pugh, S, "Total Design", Addison-Wesley Publishers Ltd., Essex, 1991.
- Swift, K. G., "Knowledge based design for manufacture", Kogan Page London, 1987.
- Yan, X.T., Borg, J.C. & Rehman, F., "Scaling up domain knowldege representation in the development of knowledge intensive CAD for pro-active Design for manufacture", In 13th International Conference on Engineering Design ICED 01, Professional Engineering Publishing, London, UK, 2001, pp. 219-226.
- Yan, X.T., Case, K. and Weston, R.H., "A Generalised Approach to the Modelling of Modular Machines", *Journal of Engineering Manufacture, Proceedings Part B of Institution of Mechanical Engineers*, Vol. 208, pp.191-203, 1994, ISSN 09544054.