DESIGNING FOR MANUFACTURABILITY IN RIVETED JOINTS

A. R. Kilhoffer

Cincom Systems, Inc.
2300 Montana Avenue
Cincinnati, OH 45211, USA

K. G. Kempf

FMC Corporation - Artificial Intelligence Center
1185 Coleman Avenue, Box 580
Santa Clara, CA 95052, USA

ABSTRACT

The study of human experts in the areas of design and manufacturing has lead to two hypotheses concerning the problem solving methods which these engineers utilize to attack difficult problems. The basis of both hypotheses is a modular approach to problem solving. One hypothesis addresses the nature of the modules utilized while the other hypothesis deals with the organization of the modules. A knowledge-based system has been designed and implemented under the philosophy expressed in these hypotheses. The domain is the design and manufacture of riveted joints in sheet metal. Special emphasis is given to the integration of design knowledge and manufacturing knowledge for the concept of "designing for manufacturability". The implementation is described in some detail and two example problems are presented with their solutions.

I INTRODUCTION

There are at least three stages in the production of mechanical goods including product and part design, process planning and scheduling, and shop floor execution. There are a variety of reasons to attempt to automate the flow of information within and between these stages, many of which have to do with increasing the quality and decreasing the cost of the goods produced, some of which have to do with the rapidly shrinking base of human experts capable of performing the necessary tasks. Application of the numeric processing power of the digital computer has already lifted part of the burden from the human engineer, making routine calculations more accurate (e.g. tolerance charting) and reducing previously impractical calculations to standard practice (e.g. finite element analysis). Although this ongoing first stage of computer automation can be considered as eminently successful, it has not solved all problems. It is often the case that a significant portion of the information flow within a particular engineering task cannot be described numerically, and so has eluded automation. More importantly in the context of this research, it is frequently the case that numeric processing does not provide assistance in automating the information flow between tasks.

In most cases it is obvious that the essential ingredient which has not been captured numerically is the experiential knowledge of the engineer. Authors have therefore suggested that the second wave of automating the information flow in design and manufacturing will be based on the techniques of Artificial Intelligence [Simmons 1984, Kempf 1985]. This assertion has been tested in some areas of product and part design [Brown and Chandrasekaran 1983, Dixon and Simmons 1983], as well as some topics in process planning and scheduling [Descotte and Latombe 1985, Phillips and Mouleeswaran 1985, Fox 1983]. From the limited amount of data available at this time, it certainly appears that AI techniques are very useful within specific engineering functions. More recently, work has been started to assess the value of AI techniques in managing and automating the process planning and scheduling functions and shop floor execution functions [Fox and Kempf 1985a, Newman and Kempf 1985, Fox and Kempf 1985b, Fox and Kempf 1986]. Once again, the preliminary results are encouraging. The purpose of the research described here is to begin an assessment of the utility of applying AI techniques to the integration of product and part design functions with process planning and scheduling functions. In the jargon of the engineer, we are concerned with the concept of "designing for manufacturability".

The specific domain of concern is the detailed design and process planning for riveted joints between sheets of aluminum alloy. From the design perspective, the engineer already knows that a joint must be present, but has not yet designed the joint in detail. The input from the design side is a rough idea of the geometry of the joint and the detailed functional requirements that the manufactured joint must exhibit. From the process perspective, the engineer already knows his resource catalog - the tools, equipment, and capabilities that exist on the factory floor - and this serves as the manufacturing input. The designer is interested in producing as output a design blueprint with detailed sheet positions including overlap and any stiffening plates required for a joint which will meet the functional requirements. The output which the process planner is seeking is a manufacturing blueprint with the type, number, and positions of the rivets which need to be installed. Our concern in pursuing this research was to ascertain whether the knowledge of the design engineer and the process engineer can be captured and utilized, and, more importantly, whether the knowledge from these sources can be integrated.

II THEORETICAL APPROACH

The fact that it usually takes a new college graduate between ten and twenty years to achieve expert status in the design and manufacturing community gives some indication of the inescapable difficulty of capturing such expertise in a computational model. Furthermore, the fact that few design and manufacturing engineers have made much progress in effectively integrating their respective knowledge makes our problem even more challenging. But these human experts still serve as our existence proof that designing for manufacturability is possible. It is through careful observation of these experts that a phenomenonological model of their reasoning methods has begun to emerge. This model is still tentative, containing errors of omission and commission, but serves as the basis for the implementation described here.

Design and manufacturing experts give the impression that they solve problems in a modular fashion. Our current model of their methods includes two major hypotheses. One hypothesis concerns the organization of the reasoning modules used while the other involves the identity of the modules included in the process. It is encouraging to note that both of these hypotheses map onto existing AI techniques.

In terms of module organization, the experts observed decompose the problems which they attempt to solve in two distinct ways (Figure 1). On one hand, they use techniques such as top down reasoning and hierarchical abstraction, solving the problem at many levels of detail, usually working from less detail to more detail (the vertical bars in Figure 1). In this process they usually have a different representation of the problem and bring different knowledge to bear at each level. On the other hand, they use techniques such as divide and conquer and cooperative problem solving. Within the same level of abstraction, using a consistent problem representation, the problem is dissected into separate but related pieces (the horizontal bars in Figure 1). These subproblems are then attacked using the appropriate knowledge for each.

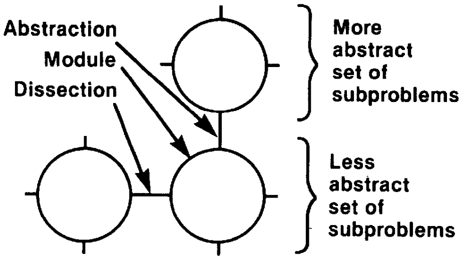


Figure 1. Basic Organization of Design and Manufacturing Expertise

In terms of module identity, the experts observed use a wide variety of distinct reasoning methods (Figure 2). Some of these modules are described here with the qualification that the list is in no way exhaustive, including only those modules which are needed to describe our implementation. Notice that while the modules presented are all described at a common level, each can be applied at any level of abstraction or dissection presented in Figure 1.

A module which is often encountered is the algorithmic module shown in Figure 2a. The basic idea here is to numerically compute a result given some data, but this often involves application of expert knowledge to select the appropriate equations or tables, to correctly insert or extract the data, and to interpret the result in the context of the problem at hand. A second commonly encountered module is the knowledge-based selection module shown in Figure 2b. Here the idea is to apply knowledge, often held as rules, to determine the heuristically best selection from a set, all the members of which might potentially suffice as solutions. The knowledge-based restriction module shown in Figure 2c is also encountered fairly frequently. In this module, knowledge, often held as constraints, is used to prune a search space with the goal of ascertaining whether any solution remains. The response set includes no, yes, and maybe, the latter two accompanied by subspaces which require further investigation.

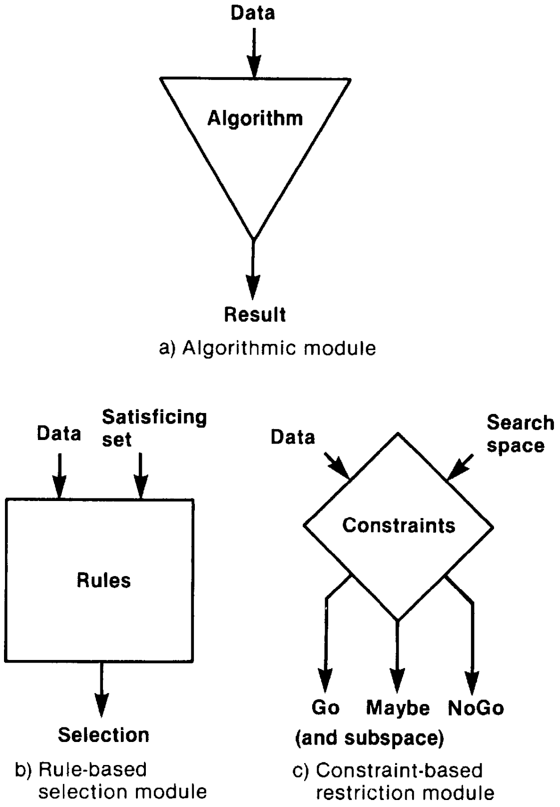


Figure 2. Some Modules of Design and Manufacturing Expertise

In summary, the first hypothesis concerns an abstraction and dissection hierarchy of problem solving activity while the second hypothesis deals with a subset of current problem solving techniques (also expressible as an abstraction hierarchy from a more global perspective). Our implementation attempts a mapping between these two structures in the context of designing for manufacturability for riveted joints.

III IMPLEMENTATION APPROACH

The implementation described here represents an initial attempt to provide intelligent automated assistance to engineers concerned with the riveting, welding, bonding (with glue), and fastening (with bolts) of aluminum, steel, and titanium sheets in various geometries. It also serves as a test for the two hypotheses outlined earlier. The subsystem described here is called REX (Riveting EXpert) and addresses the riveting of aluminum sheets. REX is implemented in VAX Common LISP (version 1.0) and runs under VMS (version 4.1A) on a VAX 8600. The current system represents roughly 350 hours of knowledge collection including direct interaction with a domain expert and study of documents provided by the expert. A few dozen example problems were solved during the collection process. The system provides useful assistance to an engineer with many years of experience in that it frees the engineer from making detailed considerations, thereby promoting a more global view of the problem at hand, and provides consistent, quality solutions quickly (roughly one minute of user input followed by one to two minutes of REX run time).

Knowledge representations in REX are composed of frames, each frame with any number of slots, each slot having value and default facets. In some cases, slots contain the attributes of objects while the value and default facets contain individual or inclusive (or exclusive) sets of reals, integers, or strings. Slots can also indicate inheritance with facet values taking on the frame names of parents and children. Values about the problem at hand are supplied by the user while the domain expert supplies values about the domain, the defaults, and inheritance. In other cases, default facets are not used, but value facets can contain executable constraints or rules supplied by the domain expert as well as textual explanation for the user and source information for system maintenance. Slots can also provide focus of attention containing frame names of related or associated constraints and rules.

The input to REX consists of two frames. One is user supplied and contains a detailed description of the problem to be solved. The contents are divided into two major sections - materials and requirements. In the materials section, information about composition is made available including metal type (currently limited to aluminum), alloy number, and temper condition. Also stored in the materials section is the length, width, and thickness of the pieces to be joined. Note that the two pieces can have different compositions and/or dimensions. Finally, the materials section contains information about the basic joint geometry (currently limited to flat joints although L-shaped and T-shaped joints have been considered) and about how many sides of the joint will be accessible during manufacturing. In the requirements section, the loads to which the finished joint will be subjected are described by a type (currently limited to axial sheer although other types have been considered) and a magnitude. Appearance requirements are also stored here and include flushness relative to sheet edges and relative to rivet heads. Finally, the requirements section contains information about the operational environment that the finished joint must withstand (currently limited to vibration although corrosion and temperature have been considered).

The other input frame is user modifiable and contains design and manufacturing criteria. The design criteria include safety margin and weight criticality information. REX uses the safety margin data to design the joint to handle worse case overloads. The weight criticality data is considered whenever there is a design tradeoff to be made and there is an option to minimize the amount of sheet overlap or minimize the number or size of rivets. The manufacturing criteria are included in two forms. On one hand there is riveting technology data about the limits of riveting in any shop regardless of facilities. Examples might include the standard length and dimensions of rivets supplied by manufacturers or the minimum rivet "drive-up" possible (the amount of shank flattened during rivet installation). On the other hand there is data about the particular facilities with which the user operates. Examples could include the minimum translational motion of automatic riveting equipment or the maximum gauge of sheet which can be bent to form joggled joints.

The output from the system consists of a precise design and manufacturing specification of the joint to be riveted. One part of the specification contains information especially interesting to the design engineer and consists of the detailed positions of the sheets including overlap and the description and location of any additional strengthening sheets that REX has decided are needed. The other part of the specification is particularly interesting to the manufacturing engineer since it contains the type, number, and position of the rivets which need to be installed.

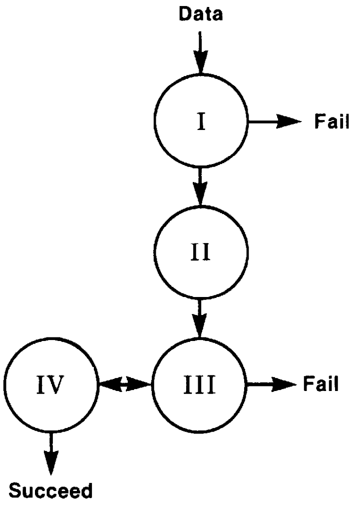
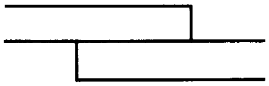
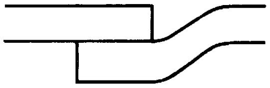


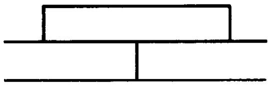
Figure 3. Organization of Riveting Expertise



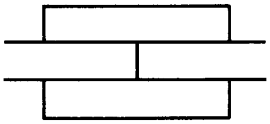
a) Simple lap joint



b) Joggle lap joint



c) Single splice butt joint



d) Double splice butt joint

Figure 4. Joint Types

REX operates at three levels of abstraction, each with its own problem representation, each with access to the problem and criteria frames. At the abstract level of module I (Figure 3), the problem is to decide the suitability of using riveting technology to produce the joint described under the criteria given. The problem representation is thus in terms of the overall limits of riveting for any joint regardless of geometry. At the intermediate level of module II (Figure 3), the problem is to design the geometry of the joint. Thus, the representation is in terms of the number and relative geometric position of the sheets involved as shown in Figure 4 for flat joints. At the detailed level of modules III and IV (Figure 3), the problem is to select and position the individual rivets. The representation here includes the detailed geometry of each sheet and is used to solve the problem by considering the sheets in pairwise fashion since all joints are arrangements of some number of simple lap joints (Figure 4a).

The most detailed level of REX shows a dissection into two cooperative problem solvers. Module III reasons over the selection of rivets of the appropriate type. Module IV reasons over the diameter and placements of the rivets in rows relative to the sheet edges. Both of the modules have access to an abstraction hierarchy of frames containing information about each of the rivets available in the production facility. The contents are arranged in seven levels with individual rivets as leaves, and with each intermediate node expressing the range of properties of all of the rivet instances below it. Intermediate levels classify by such criteria as access type (one side/two sides), head type (flush/protruding), shank type (solid/tubular), and rivet material.

The overall control among the modules is by simple serial invocation. Module I is activated after the user closes the input files. If module I fails, control is passed back to the user along with an explanation of the difficulty. Otherwise control is passed to module II, although the user may be notified if module I has detected that the solution to the problem is close to the limitations of riveting technology. After module II has completed its reasoning, control is passed to module III along with a description of the joint geometry of choice. Module III, if it can not find a rivet which it considers to be a good candidate, reports failure with an explanation to the user. Otherwise it passes control and a description of the rivet selected to module IV. If the rivet suggested can be placed so as to solve the problem, the user is notified of the details of the solution. If module IV fails in its attempt to place the rivet, control is returned to module III with a request for an alternative suggestion.

Considering the individual components, module I (Figure 3) is a knowledge- based restriction module (Figure 2c). The problem and criteria frames supply the input. The search space implicitly contains all imaginable joints regardless of their manufacturability. A basic model of riveting is expressed in a network of constraints represented in frames. Values from the criteria frame parameterize the model while values for the current problem are extracted from the problem frame. Module I compares the key problem requirements with the crucial limitations of the technology. Each such comparison results in an indication (go, maybe, nogo) and an explanation (for maybe and nogo). This module attempts to decide whether the stated problem lies inside or outside of the set of manufacturable joints, or is close to the border. It is here that the system holds knowledge that, for example, a joint cannot be flush relative to sheet edges on both sides, or that it is not a good idea to try to rivet foil. This module has the power, for example, to understand that a flat joint with a one sided edge flushness requirement between sheets of different gauges will need one sheet to be joggled, but that the thinnest sheet being over the maximum gauge which the shop can bend will preclude manufacture. It can also explain this to the user.

Module II (Figure 3) is a knowledge-based selection module (Figure 2b). The problem and criteria frames supply the input. The satisficing set for flat joints contains the arrangements shown in Figure 4. A basic model for making a selection is expressed in a set of rules represented in frames. Module II reasons (mainly) over the load and appearance criteria with strong influences from (mainly) the manufacturing access and weight restrictions to select the appropriate basic joint design in the context of the problem at hand. For example, the joggle lap joint presents a flush appearance without weight penalty, but is difficult to manufacture. The double splice joint can carry high loads, but adds significant weight and requires double sided access for easy manufacture.

Module III (Figure 3) is a hybrid between knowledge-based restriction (Figure 2c) and knowledge-based selection (Figure 2b). The input to both is supplied from the problem and criteria frames along with the joint type passed from the previous module. The search space for the restriction section is the rivet frame. The goal of the constraints contained in this section is to prune all the rivets which will not work for the problem of concern. The satisficing set for the selection section is the set of remaining rivets. The goal for the rules contained in the selection section is to rank the rivets in the context of the problem. The best candidate is selected and passed to module IV, but the ranked list is maintained in case module IV fails and asks for another selection. Module III fails when its list is empty and module IV has not yet produced a satisfactory solution. It is often the case that module III reasons about ranking a large number of rivets. This is because the problem characteristics do not always allow the restriction section to prune much from the rivet frame. For example, a flushness requirement relative to rivet heads will eliminate further consideration of protruding rivets, but the lack of a fine flushness requirement does not allow flush rivets to be pruned.

Module IV (Figure 3) is an algorithmic module (Figure 2a) containing three algorithms and knowledge in the form of frames about the algorithms. Input comes from the problem frame, the criteria frame, and the rivet frame along with the rivet passed from the previous module. The first algorithm is concerned with selecting a rivet diameter and calculating the rivet spacing. This task utilizes lookup in tables associated with rivet instances and requires rules for table and entry selection. The second algorithm distributes the rivets into rows, but since various tradeoffs need to be made here, rules are associated with this distribution process. The final algorithm checks the rivet and row spacing to verify load carrying capacity in light of the safety margin. Success of the last algorithm triggers a success report to the user. Failure causes a request to be sent back to module III for another candidate.

At this point a fifth module could be added to complete the manufacturing blueprint in terms of clamping the sheets, drilling and countersinking the holes including deburring, and inserting and seating the rivets. While this module has not yet been implemented, initial studies indicate that it will be a knowledge-based selection module over manufacturing equipment and will include rules for sequencing the selected manufacturing operations.

IV VERIFICATION

One human expert provided all of the domain knowledge about riveting which was needed to bring REX to a prototype stage. Various trace and debug facilities which REX contains were used as well as consulting sessions with the domain expert to verify the initial system performance. At that point, a second domain expert from a different division was contacted to provide an independent test of the system.

Two approaches were taken to obtain adequate test coverage. One set of four simpler problems was solved by REX and by the second expert independently and the answers were compared. A second set of four harder problems was solved by REX and the second expert was asked to review and comment on REX's solutions. The input and REX's output for one problem from the first set are presented in detail. This problem is of interest because it is the only problem from either set for which REX's solution was not verified by the second expert. The input and REX's output for one problem from the second set are also presented in detail. This problem is the most difficult problem in the entire set and demonstrates that REX is capable of solving problems of moderate complexity. Note that the numeric accuracies quoted reflect manufacturing data in the criteria data frame.

In the first problem, one sheet is a 36" long, 12" wide, 0.25" thick piece of 2024-T6 alloy and the other sheet is a 24" long, 12" wide, 0.1875" thick piece of 2024-T86 alloy. Manufacturing access can be gained from only one side of the joint and there is a one-sided sheet edge flushness requirement but no rivet head flushness requirement. The load is relatively low at 1500 pounds.

REX correctly selected a joggle lap joint, a NAS1398D one-sided rivet with an 0.156" shank diameter and an 0.5" shank length, and placed a single row of rivets with an 0.3125" rivet-sheet edge spacing. Unfortunately, REX suggested that only three rivets were required with a 5.6875" rivet-rivet spacing. The expert was quick to point out that while that might be enough rivets to handle the load, the spacing between rivets was too large and the sheets might buckle when load was applied. On reflection, the system does contain knowledge about the minimum allowable spacing between rivets since the original focus was on heavy loads, but no knowledge about the maximum allowable spacing. The expert contributed the heuristic that sixteen times the rivet diameter should be used as the maximum, a chunk of knowledge easily incorporated into REX.

In the second problem, one sheet is a 36" long, 12" wide, 0.375" thick piece of 2024-T6 alloy and the other sheet is a 24" long, 12" wide, 0.25" thick piece of 2024-T86 alloy. Manufacturing access can be gained from both sides of the joint and there is a one-sided rivet head flushness requirement but no sheet edge flushness requirement. The load is relatively high at 45000 pounds.

REX correctly selected a single splice butt joint and a NAS20426AD two-sided rivet with an 0.156" shank diameter and an 0.75" shank length. REX suggested that 88 rivets were required in seven rows. Rows one, three, five, and seven hold 13 rivets each while rows two, four, and six hold 12 rivets each. The layout includes a rivet-sheet edge spacing of 0.375", a row-row spacing of 0.75", and a rivet-rivet spacing of 0.9375".

V CONCLUSIONS

Although the two hypotheses put forward concerning the problem solving methods of human design and manufacturing engineers are simplistic, the prototype system designed and implemented under the philosophy expressed in these hypotheses appears to be reasonably useful. Both design and manufacturing expertise has been captured and integrated under the concept of design for manufacturability. The combined expertise has been utilized to provide solutions which have been validated not only by the expert who contributed the initial knowledge to the system, but also by a second independent expert. However, as with any AI system, the real test of the ideas lies in the future as the prototype is expanded and extended towards a tool for daily use by engineers.

ACKNOWLEDGEMENTS

The work described here is extracted from the Masters thesis of A. Kilhoffer at the University of Missouri and owes much to the support of Prof. D. St Clair. The help of George Palcheff as our primary domain expert and of Richard Stover as our verification expert are gratefully acknowledged.

REFERENCES

1. M. K. Simmons, "Artificial Intelligence for Engineering Design", Computer-Aided Engineering Journal, 1, 3, pp. 75-83, 1984.
2. K. G. Kempf, "Manufacturing and Artificial Intelligence", Robotics, 1, 1, pp. 13-26, 1985.
3. D. C. Brown and B. Chandrasekaran, "An Approach to Expert Systems for Mechanical Design", Proc. Automating Intelligent Behavior : Applications and Frontiers (Nat. Bureau Stds., Gaithersburg, MD), pp. 173-180, 1983.
4. J. R. Dixon and M. K. Simmons, "Computers that Design: Expert Systems for Mechanical Engineers," Comp. Mech. Eng., 2, 3, pp. 10-18, 1983.
5. Y. Descotte and J.-C. Latombe, "Making Compromises among Antagonist Constraints in a Planner", Artificial Intelligence, 27, 2, pp. 183-218, 1985.
6. R. H. Phillips and C. B. Mouleeswaran, "A Knowledge-Based Approach to Generative Process Planning", Proc. AUTOFACT '85 (Detroit, MI), Sect. 10, pp.1-15, 1985.
7. M. S. Fox, Constraint Directed Search : A Case Study of Job-Shop Scheduling, Ph.D. dissertation, Computer Science Dept., Carnegie-Mellon University, Pittsburgh, PA, October, 1983.
8. B. R. Fox and K. G. Kempf, "Opportunistic Scheduling for Robotic Proc. Assembly," IEEE Inter. Conf. Rob. Auto. (St. Louis, MO), pp. 880-889, 1985a.
9. P. A. Newman and K. G. Kempf, "Opportunistic Scheduling for Robotic Machine Tending," Proc. IEEE Conf. AI Appl. (Miami Beach, FL), pp. 168-175, 1985.
10. B. R. Fox and K. G. Kempf, "Complexity, Uncertainty, and Opportunistic Scheduling", Proc. IEEE Conf. AI Appl. (Miami Beach, FL), pp. 487-492, 1985b.
11. B. R. Fox and K. G. Kempf, "A Representation for Opportunistic Scheduling," Proc. 3rd Inter. Symp. Rob. Res. (Gouvieux, France, 1985), to appear, 1986.