THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS
345 E. 47 St., New York, N.Y. 10017

87-GT-217

Initial User Experience with an Artificial Intelligence Program for the Preliminary Design of Centrifugal Compressors

CAROL J. RUSSO
General Electric Co.
Aircraft Engine Business Group
Lynn, Mass. 01910

DENNIS J. NICKLAUS AND SIU S. TONG
General Electric Co.
Corporate Research and Development
Schenectady, NY 12301

ABSTRACT

A new approach is evaluated for the design of turbomachinery components using existing analysis codes coupled to a generic Artificial Intelligence (AI) software framework called ENGINEOUS. This AI framework uses intelligent search techniques with a small set of basic component design rules to iterate to an optimized solution and to quantify parameter trade-offs. Initial experience with ENGINEOUS indicates that it is a powerful design tool which quickly identifies non-obvious solutions balanced for conflicting multiple goals in a small number of iterations which vary linearly with the number of variables. The solution path and driving logic are easily visible to the designer and a parameter study option can rapidly quantify potential design trade-offs which together allow a critique of the selected design to balance performance against development risks. Because this AI design approach fosters intelligent interface with the designer and is generic, the potential application areas and productivity benefits appear enormous.

NOMENCLATURE

D	impeller relative diffusion factor
MTIP	impeller exit absolute Mach number
M1 relative	impeller leading edge tip relative Mach number
NB	number of impeller blades
PR	total-to-total centrifugal stage pressure ratio
PT	impeller inlet total pressure
R*CU	product of radius and absolute tangential velocity
RHUB	impeller leading edge hub radius
RPM	impeller physical rotational speed
RTIP	impeller exit radius
TT	impeller inlet total temperature
UTIP	impeller exit tip speed
WC/A	corrected inlet airflow/inlet area
α	impeller exit absolute flow angle
$\alpha 1$	impeller inlet absolute flow angle at root-mean-square radius
β	backsweep or impeller exit relative flow angle
Δ	change in any quantity
η	total-to-total centrifugal stage adiabatic efficiency

BACKGROUND AND INTRODUCTION

Reduced development cost and time are a significant competitive advantage in today's jet engine market and have motivated many recent advances in turbomachinery component design techniques. The increasing demand for improved performance with lower development risks, however, has multiplied the iteration time needed to balance conflicting aerodynamic and mechanical goals and complicated the design process. The essence of the design process is not to produce a mathematically optimized design from some "black-box" series of programs but to identify a design that is balanced for the set of conflicting goals generic to the particular application under study and to quantify potential parameter trade-offs so that the final design will have the highest performance possible with acceptable development risks. Therefore, any automation of the design selection process must produce a logical path visible to the design engineer in order that trade-offs can be quantified and reviewed against the original requirements of the application to gain confidence in the final design selection and to properly balance aero and mechanical performance against development risks. Because of the critical requirements of intelligent interaction with the designer and resolution of complex and conflicting goals, the use of recent but well-developed software systems (1,2) for applying artificial intelligence (AI) to engineering problems appeared very attractive.

Presented at the Gas Turbine Conference and Exhibition, Anaheim, California — May 31-June 4, 1987

In the past three years, many papers have appeared on applying AI techniques to engineering problems especially in the aerospace and automotive industries (3,4,5,6,7,8,9,10,11,12). Many of these applications involve developing expert design systems such as EXFAN (13) for the design of cooling flow fans or LOKI (14) for the design of aircraft wings in which all of the information needed to design the component is encapsulated in a series of rules. This approach is not an attractive one to apply to the design of turbomachinery components for several reasons. First for complex components, these expert systems require many man-months to develop for each component. In addition, the physical modelling in analysis codes is necessarily limited and it is difficult to write a complete set of rules which also compensates for modelling deficiencies. More importantly, the complex relationships among aero and mechanical parameters for many turbomachinery components cannot be encapsulated into a complete or nearly complete system of rules that applies for the broad range of jet engine applications and that includes the high degree to which technical judgement must often be applied to balance performance against development risks.

On the other hand, entirely discarding the human understanding of the primary physical relationships and the empirical design knowledge base to rely entirely on purely mathematical and logical search and optimization techniques is also an unattractive approach. This approach is computationally intensive since the analysis code must be run once for each parameter to calculate its derivatives relative to the optimization parameter in order to select a search direction with periodic re-runs to recalculate these derivatives. Therefore, the computational time required to reach a numerically optimized solution increases exponentially with the number of parameters. Also, there is critical information about the relationships among various key parameters in the analysis codes and about logical paths to select and iterate among analysis codes that is very difficult to express in conventional sequential programming languages. Finally, the numerical approaches are too rigid to easily implement backtracking to overcome analysis code floating point errors which can occur frequently for advanced turbomachinery designs.

Therefore, a new approach was adopted which utilizes a subset of key component design knowledge captured in a frame/rule based system which guides a collection of intelligent search techniques to explore the design space in an effective manner. This AI system mimics the educated trial-and-error design iteration process of an experienced designer without using artificial convergence or damping parameters. Confidence in the final design can be high because the solution path and driving logic are easily visible to the user and because a parameter study option quickly quantifies potential trade offs. Modifying and selecting the final design by assessing the proper balance between performance and development risk can then be quickly accomplished by an experienced engineer. In addition, because this AI design system contains facilities to easily capture component or even application dependent design knowledge and supplement it with problem independent intelligent search techniques, the AI shell is generic and can be coupled to any analysis code or group of codes.

To assess the effectiveness of this approach and the advantages and disadvantages of applying AI to the design of turbomachinery components, an AI software shell for intelligent seach and optimization, called ENGINEOUS, has been developed around a set of existing aerodynamic and mechanical centrifugal stage one-dimensional (1-D) design programs. One dimensional programs were selected because they are generally less complex than detailed two- or three-dimensional programs but still enable the resolution of conflicting goals from multiple disciplines.

This paper reports on the initial user experience with ENGINEOUS coupled to the Fortran 1-D aerodynamic code CENTCAL and its companion geometry generating code GEOCENT. The AI software details of ENGINEOUS and the coupling of multiple analysis modules are discussed in reference 15. A 1-D mechanical impeller disk analysis program DISKSHAPE has been coupled to CENTCAL and is currently being extensively exercised. Initial experience with the program CENTCAL has included the resolution of conflicting multiple goals similar to those encountered with the addition of a mechanical disk analysis code. However, the experience reported in this paper involves only one analysis code which was sufficient to demonstrate the potential benefits of this AI design approach.

GENERAL PROGRAM COMPOSITION

Solution Techniques

ENGINEOUS combines a partial, qualitative knowledge base with advanced searching heuristics in order to explore a design space and to optimize a design. The knowledge base is specific to each analysis code and is comprised of rules such as:

"To increase efficiency, increase backsweep."

which are listed in order of effectiveness or preference. Rules can be expressed conditionally and applied selectively. The knowledge base for the centrifugal stage analysis code CENTCAL has a total of 46 rules, 25 basic rules and 21 inverse rules. These rules do not completely define the design space so ENGINEOUS adds intelligent search heuristics to enable the program to:

- o avoid local optima
- o explore variable changes where dependencies are not given in the knowledge base
- o combine rules to change more than one variable at a time
- o sort proposed variable changes in order of "most likely to improve the design"
- o backtrack around input values that cause the analysis program to "bomb".

In this way, a small set of basic rules can be amplified to powerfully drive an optimization process avoiding the problems of attempting to construct a complete expert system for designing complex turbomachinery components. In essence, ENGINEOUS reduces the number of possible variables to a much smaller subset using human knowledge of what parameters are most important under which conditions. A more complete description of the AI search techniques in ENGINEOUS can be found in reference 15. ENGINEOUS is written in LISP (16) and uses the knowledge based system shell KEE (17).

User Interface

ENGINEOUS has a flexible, menu-driven, and user-friendly interface for setting up and performing one or more of the following three basic functions using a given analysis code and rule base:

Mode A
- o bring design within default or user-selected constraints and/or

Mode B
- o optimize a design with user-specified variables and goals and/or

Mode C
- o perform a single or multiple parameter study with or without optimization at each step.

These choices are menu-driven with ENGINEOUS handling the required internal manipulations. Complex problem set-up structures such as multiple parameter studies with optimization to multiple goals at each step can be saved for repeated use on different design applications. A user can start a given design to meet a set of production, development, or advanced application specific set of constraints, and then optimize the design for a given goal. The user can then quickly quantify key parameter trade-off choices by performing a pre set or user-specified multiple-parameter study around the optimized design where each step of the parameter study is optimized for other variables within constraints. A very rapid and quantified picture can therefore be obtained for the relevant design space for a given application.

Once the selection of analysis code and primary rule base is made, rules can be added for a given application by answering simple questions without a detailed knowledge of KEE or LISP. Considerable effort was extended to avoid "black-box" programming by providing easy access to the solution path and the primary driving design parameters for a particular application both during and after the ENGINEOUS run. In addition, the parameter study option allows the user to see where in the design space any particular ENGINEOUS solution lies so that the original assumptions and goals of the design can be re-assessed. The ENGINEOUS solution can then be adjusted if desired and confidence in the final selected design can be high.

Aero Analysis Code

The aero analysis codes, CENTCAL and GEOCENT, estimate the one-dimensional (1-D) performance of a centrifugal stage and construct a preliminary meridional flowpath plot. The 1-D performance code CENTCAL requires a few basic input values such as stage pressure ratio, inlet conditions, RPM, and a few exit conditions such as impeller exit absolute and relative flow angles and stage discharge Mach number. Semi-empirical loss models are used to iterate on the impeller and diffuser designs required to achieve the input stage pressure ratio. These loss models include the effects of variable gas properties, impeller inlet absolute and relative Mach numbers, skin friction, blade loading, flow blockages (especially at the diffuser inlet), impeller exit absolute Mach number, relative and absolute diffusion, blade clearances, and Reynolds' number. An inlet file is written to GEOCENT which calculates the remaining impeller and diffuser geometries.

These analysis codes have been extensively compared to in-house test data over a wide range of pressure ratios for both production and development centrifugal stages and yield realistic preliminary design performance and geometry numbers. The codes are equipped with warning messages if certain design parameters fall outside of the current range of experience but the solution is not blocked. These codes have proved to be very useful in quickly assessing the performance and geometry implications of various basic design choices.

INITIAL USER EXPERIENCE

A matrix of over 100 test cases, many from existing engine designs and studies, has been run to compile initial experience with ENGINEOUS. The matrix was configured primarily to address the following issues:

Modes A and B:
- o Does ENGINEOUS arrive at the same optimized solution with widely different initial starting points?
- o Does ENGINEOUS repeatedly drive to solutions that are at obvious parameter limits for simple goals?
- o How does the ENGINEOUS solution compare to well-iterated "expert" solutions?

Mode C:
- o How useful is the option to optimize at each step of a specified parameter study?
- o How does the design space which has been "optimization by hand" at each step compare to the design space given by ENGINEOUS?

General:
- o How does the total number of iterations behave as the number of variables and goals are increased?
- o How does the ratio of "good" iterations to total iterations behave as the number of variable and goals are increased?

The results of the test cases regarding each of these issues are discussed in the following sections.

Experience with Mode A (Satisfying Constraints) and Mode B (Finding Optimized Solutions)

ENGINEOUS is set up with default input parameter values which are well within the empirical data base of the aero analysis code CENTCAL. When these default values were used, ENGINEOUS was able to modify the specified variable input parameters to satisfy the specified dependent constraints within 3-6 iterations and then proceed to optimize the solution. Several cases were also run with some initial input parameters such as backsweep set to values at the high and low ends of the range of experience and ENGINEOUS drove to the same solution within ±0.1 points in efficiency.

Because the rule base was small and limited to relatively basic physical behavior, many cases were analyzed to determine if ENGINEOUS would simply drive to an obvious parameter limit. As one would expect, for the single "max η" goal with low to moderate pressure ratios and with only the strong variables α, β, and RHUB, ENGINEOUS drives the backsweep up to lower the impeller exit M and the resulting diffuser inlet losses and drives the inlet hub radius down to lower inducer relative M losses. However, if other input parameters are allowed to vary (within constraints), the limits on backsweep and RHUB are often not reached and the solution is a complex balance among 5-6 relatively strong variables. In addition, as soon as a second goal is added such as minimizing tip speed (to simulate mechanical disk stress constraints), the solution is non-obvious and complex.

Non-obvious solutions were also obtained for a simple "maximize efficiency" goal for higher pressure ratio cases where the design had also been well-iterated by an experienced centrifugal compressor designer. Table 1 compares the ENGINEOUS optimized solution for two well-iterated designs allowing only α and β to vary to simplify the comparison.

Table 1
Comparison of Well-Iterated Designs "By-Hand" and ENGINEOUS Solutions for "Max η" Goal

Parameter	"By-Hand" Design	ENGINEOUS Design
Case A		(30 Runs)
PR	Base (>4:1)	Base
Δη, points	Base	0
Δα, Deg.	Base	-3.8
Δβ, Deg.	Base	-5.6
Δ U_{tip}, FPS (M/S)	Base	+10 (+3.0)
D	0.493	0.413
Δ Impeller exit M,%	Supersonic	+1.9
Case B		(23 Runs)
PR	Base (>4:1)	Base
Δη, points	Base	+0.7
Δα, deg.	Base	+0.3
Δβ, deg.	Base	+5.2
$ΔU_{tip}$, FPS (M/S)	Base	+29 (+8.8)
D	0.440	0.413
Δ Impeller Exit M,%	Supersonic	-2.8

These two cases show that the ENGINEOUS solution was equal or higher in efficiency at a significantly lower impeller diffusion factor (a prime aero risk factor). The lower diffusion factor solution was not initially obvious to the experienced designer because the pressure ratio was higher than previous designs and some compromise in diffusion factor was expected. Table 2 shows other ENGINEOUS solutions for the second design in Table 1 when a "minimize tip speed" goal was added and when other parameters were allowed to vary. The "min UTIP plus max η" goal achieved the same efficiency gain as the "max η only" goal but at a lower tip speed (prime mechanical risk factor). The large number of variable parameters gained a total of 2.3 points over the "well-iterated by hand" design and achieved an even lower tip speed. This last ENGINEOUS solution quickly quantified the trade-off between inlet hub radius and impeller exit tip speed which were conflicting requirements of this design because the preceeding axial compressor required a high exit radius to achieve high loading per stage.

TABLE 2

Multiple Goal ENGINEOUS Solutions

Parameter	"By-Hand" Design	ENGINEOUS Designs		
Goals	Max η	Max η	Max η+ MinUTIP	Max η+ MinUTIP
Variables	α β $α_1$ NB, RHUB Wc/A	α β	α β	α β $α_1$ NB, RHUB, Wc/A
No. of Runs		23	27	41
PR	Base(>4:1)	Base	Base	Base
Δη, points	Base	+0.7	+0.7	+2.3
Δα, Deg.	Base	+0.3	+1.0	+1.2
Δβ, Deg	Base	+5.2	+3.0	-1.0
Δ UTIP, fps (m/s)	Base	+29 (+8.8)	+5 (+1.5)	-34 (-10.4)
D	0.440	0.413	0.452	0.410
Δ Imp. Exit M, %	Super-sonic	-2.8	-0.9	-0.9
Δ RTIP, in. (cm)	Base	+.13 (+.33)	+.02 (+.05)	-.29 (-.74)
Δ RHUB, in. (cm)	Base	Base	Base	-1.47 (-3.7)
Δα1, Deg.	Base	Base	Base	-1.5
M1 Relative	0.80	0.80	0.80	0.70
Δ Wc/A	Base	Base	Base	0
ΔNB	Base	Base	Base	0

Experience With Mode C (Parameter Study Option)

As previously discussed, ENGINEOUS has a built in option to allow a multi-variable parameter study to be run with or without optimization at each step. Several 3x3 parameter studies were run. Figure 1 shows some results of a study on backsweep for an axi-centrifugal compressor for three sets of centrifugal stage PR, inlet PT and TT where the preceeding axial PR was varied to keep the overall compressor PR constant. The parameter study required optimization at each step.

Figure 2 compares the ENGINEOUS optimized parameter study results for a "max η" goal to the results "optimized by hand" which took an experienced designer about one day to complete the nine cases required iterating the input parameters by hand to maintain reasonable impeller diffusion factors for all cases. The ENGINEOUS optimized solution shows much tighter boundaries indicating that ENGINEOUS carried the design iteration process further than the hand optimization to obtain lower diffusion levels and higher stage efficiencies.

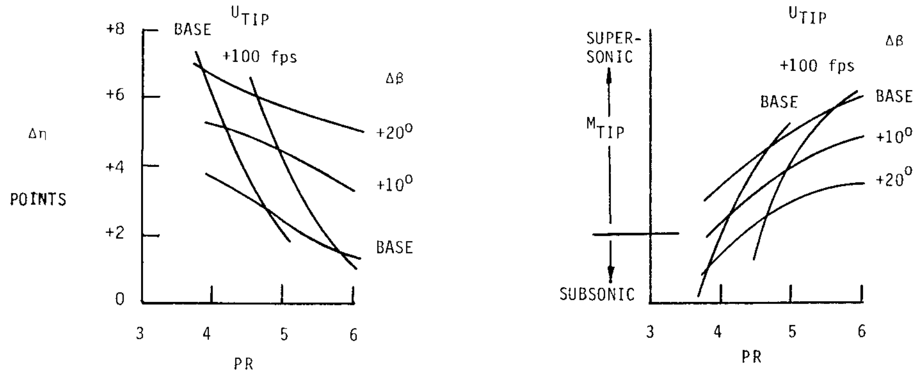

FIG. 1 PARAMETER STUDY EXAMPLE

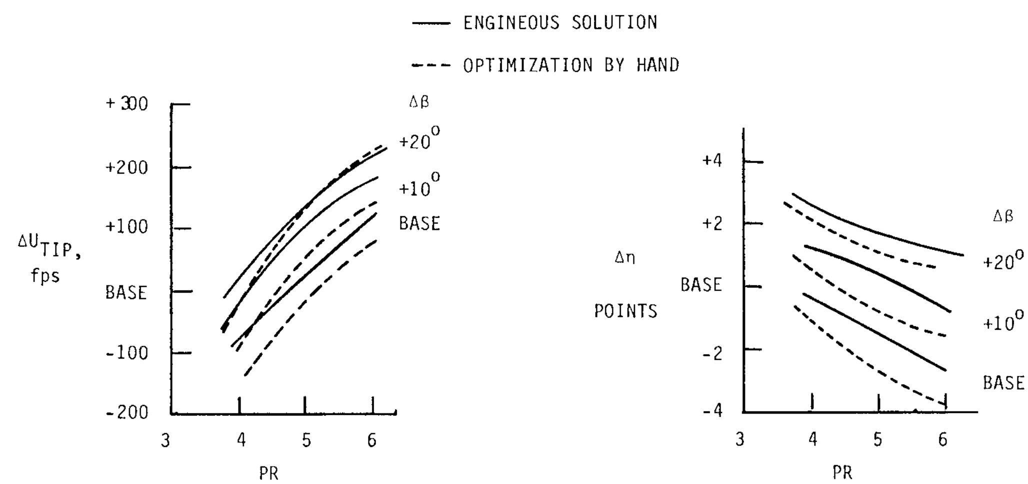

FIG. 2 OPTIMIZED PARAMETER STUDY COMPARISON

Optimization and meeting constraints at each step of a parameter study can yield very different trends from non-optimization at each step as shown in Figure 3 where the centrifugal PR, inlet PT and TT were held constant and the inlet hub radius was varied with and without optimization. The non-optimized study shows a steeper trend of η vs RHUB because backsweep alone was forced to track the hub radius and hence the inlet R*CU change to maintain constant centrifugal stage PR. The optimized solutions, however, allowed both the absolute and relative impeller exit angles to change to accomodate the inlet R*CU change and to optimized stage efficiency. The higher non-optimized efficiency solution for the lowest hub radius is not a viable solution since the impeller diffusion factor is unrealistically low which in this case implies excessive tip speed and a heavier, lower life impeller.

The parameter study option with optimization can also be used to see where in a design space a given ENGINEOUS optimized solution lies. Figure 4 shows the results of a parameter study on β, RHUB, and α, performed allowing α and β (when not specified) to vary. The locus of optimized points are plotted for each of these three parameters. The goal was "max η and min UTIP" so that the ENGINEOUS solution falls below the global peak η point. These plots can be used to quantify important parameter trade-offs and to make a more informed final design selection.

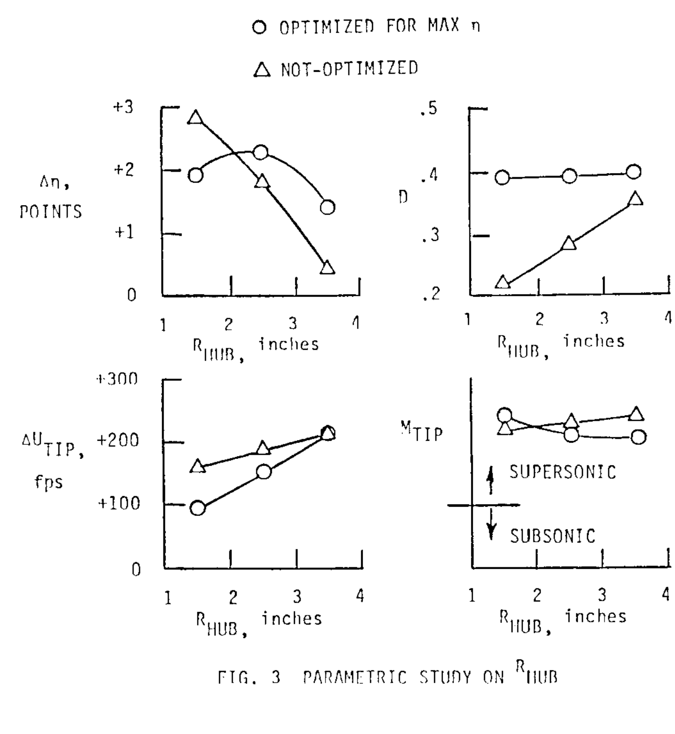

FIG. 3 PARAMETRIC STUDY ON R_{HUB}

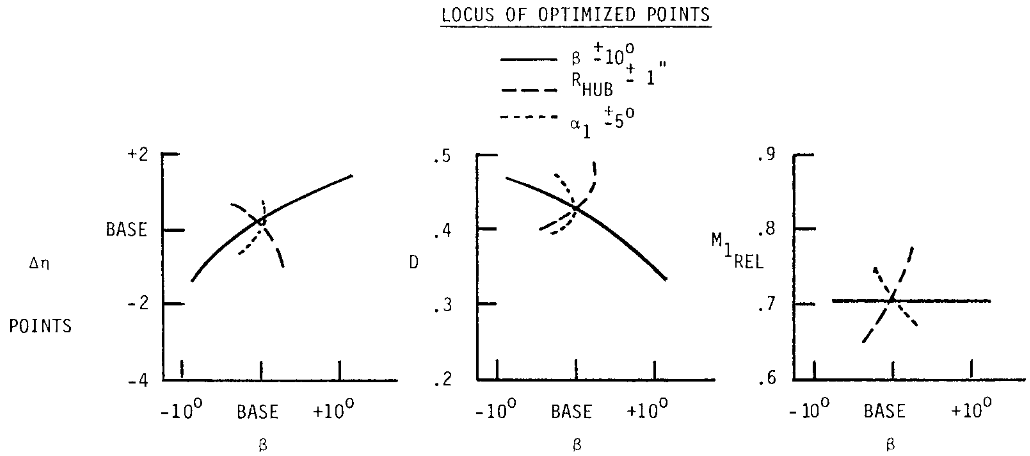

FIG. 4 DESIGN SPACE OPTIMIZATION

General Experience

Some general statistics have been compiled on ENGINEOUS from the initial matrix of test cases. Figure 5 shows the total number of iterations run, including those needed to bring the design within constraints, vs the number of variables. For those cases with the single goal of "maximize efficiency", the ratio is about 9 iterations per variable. For the dual goal "max η + min UTIP", the ratio is about 14:1. For the small sample of triple goal cases, "max η, min UTIP, and max RHUB", the ratio fell back down to about 9:1. Also shown in Figure 5 is the fact that adding weaker variables such as the number of impeller blades does not increase the number of iterations which is to be expected since ENGINEOUS concentrates its iterations on the strongest variables first. The double symbols at 4 and 5 variables show the results when two variables of approximately equal power were swapped resulting in nearly the same number of iterations needed.

Figures 6 and 7 show the ratio of "good" iterations to total iterations vs the number of variables which is henceforth called the Completeness Index CI because the higher the value of CI, the more complete the rule base. A "good" iteration is one that moves towards the goal from the previous iteration. For the single goal of "max η", the ratio of good to total iterations varies primarily between .30 and .50. The dual goal ("max η and min UTIP") ratio varied primarily between .45 and .55. The few triple goal cases fell around .35 and .40. These general statements do not appear to hold for goals involving PR probably because PR is the strongest driver on η but had the fewest rules. If more rules are added especially on key variables such as PR and α, the CI will increase although never equal 1.0 since ENGINEOUS requires some downward steps to avoid local optima.

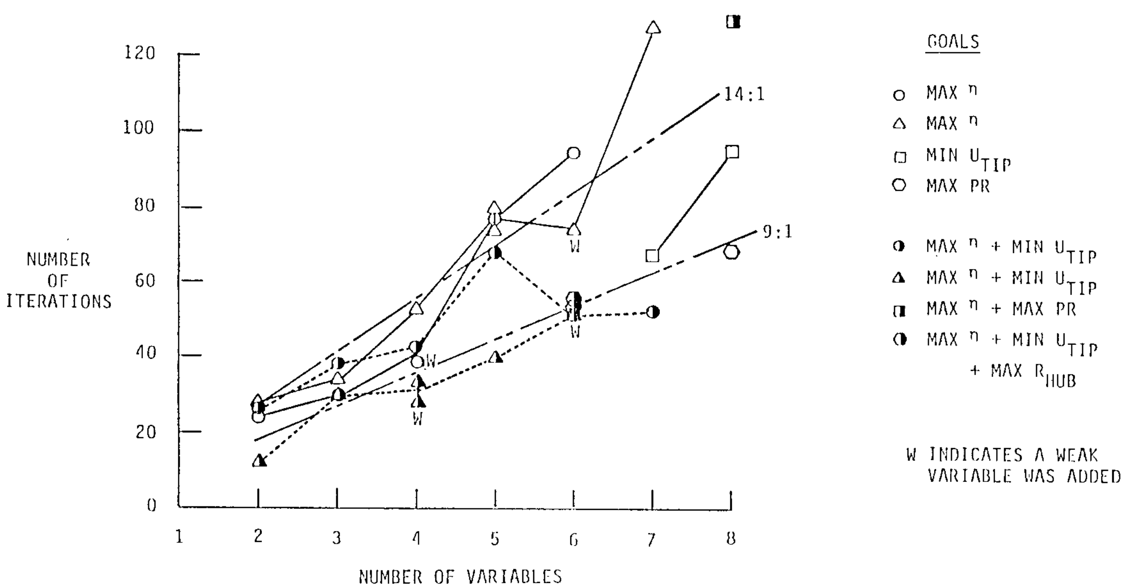

FIG. 5 TOTAL NUMBER OF ITERATIONS

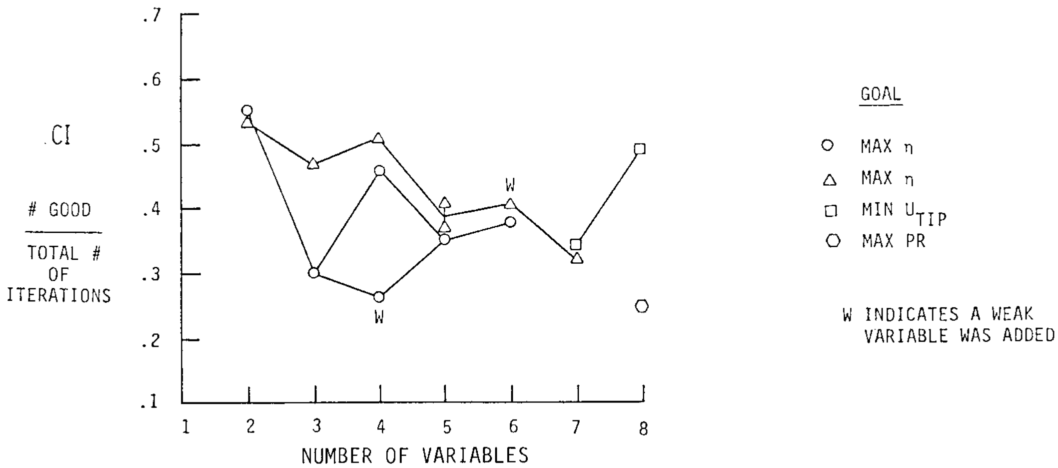

FIG. 6 COMPLETENESS INDEX FOR SINGLE VARIABLE GOALS

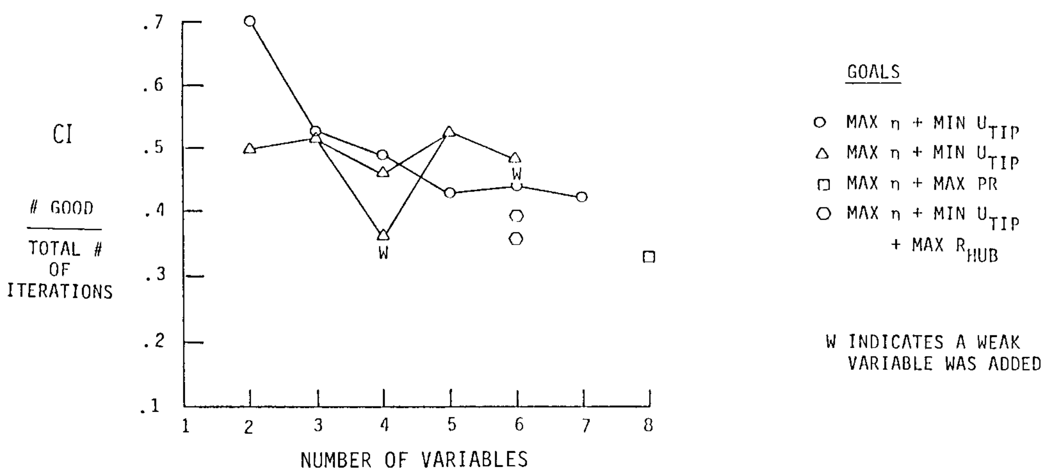

FIG. 7 COMPLETENESS INDEX FOR MULTIPLE VARIABLE GOALS

CONCLUSIONS

Some major advantages of using AI techniques to explore the application specific design space and to optimize a given turbomachinery component design are summarized below:

- The user interface is very flexible and powerful relieving the user from any requirement to know symbolic programming languages such as LISP.
- A simple and incomplete rule base was very powerful in arriving at optimized solutions with conflicting goals when coupled with the heuristic AI search techniques.
- The solution path and key quantitative parameter relationships are visible both during and after the optimization allowing the user to gain confidence in the solution and to apply more qualitative judgements such as development risk.
- Many more design options and parameter trade-offs can be explored in a given period of time.
- An inexperienced designer can quickly gain an understanding of the parameters driving a particular design without lengthy trial-and-error runs.
- ENGINEOUS can improve on designs iterated by an experienced designer particularly for unfamiliar applications.
- Application specific design rules can be easily added without reprogramming.
- The shell can couple design codes from multiple disciplines such as aero and mechanical to effectively balance inter-related and often conflicting goals.
- ENGINEOUS can easily recover from a run that "bombs" the analysis code and can avoid local optima.
- The AI search and optimization techniques are efficient for 1-D analysis codes with the number of iterations varying linearly with the number of variables.

At present, there do not appear to be any major disadvantages in applying AI search techniques coupled with a simple rule base to preliminary design of some turbomachinery components. However, there are some potential problem areas worth noting. For one, the AI search techniques can stop prematurely if there is not enough logic included to adequately explore the design space. There is not the mathematical assurance of numerical optimization techniques that at least a local optimum has been reached. However, the parameter study option can easily be used to explore the design space surrounding the AI solution to find where in that space the current solution lies. For another, applying the AI shell to analysis codes with long computational times such as 3-D viscous programs will probably be more inefficient than direct input manipulation by the design expert unless one can develop and validate sophistocated rules for searching patterns and deducing the required changes in blade or flowpath shapes. Finally, effectively using the AI shell requires the use of powerful workstations and the acquisition of software licenses for each workstation or device where the AI program resides. The cost of properly supplying a large design staff can be significant although improvements in productivity and in the quality of the designs produced should more than offset the hardware and software acquisition and support costs.

In retrospect, there appears to be a strong case for significant productivity and design quality improvements from applying AI techniques to at least the preliminary design of turbomachinery components. As is usually the case with complex tools, careful attention must be given to how and what AI tools are applied and how the designer interfaces with these tools to avoid the dangers discussed by Prof. Smith in reference 18. The easy user access to the solution path and driving logic should avoid many of the problems encountered with "black-box" programming approaches. Moreover, the current approach relieves the designer from tedious repetitive iterations and focuses attention on the most challenging aspects of engineering design; the creation of a design that resolves conflicting goals and the quantification of potential trade-offs for given applications.. More experience needs to be gained in using AI design shells like ENGINEOUS to fully assess the contribution of AI to engineering component design and to understand how best to mix the use of AI and numerical optimization techniques for different engineering design problems. The productivity potential of the current approach is clearly greatest when the rule base is focused on those parameters which typically drive the component design and when complex and conflicting multi-disciplinary goals must be simultaneously met. The generic nature of the shell and the large number of analysis codes used in turbomachinery and other engineering design fields provide a vast number of potential applications for this design tool.

REFERENCES

1. Cross, George R., "Tools for Constructing Knowledge-Based Systems", Optical Engineering, Vol 25, No 3, March, 1986, pp. 436-444.

2. Kinnucan P., "Software Tools Speed Expert System Development", High Technology, March, 1985, pp.16-20.

3. Schindler, Max, "Technology Report: Artificial Intelligence Begins to Pay Off with Expert Systems For Engineering", Electronic Design, Aug., 9, 1984, pp. 104-146.

4. Kidwell, G., and Eskey, M., "Expert Systems and Their Use in Augmenting Design Optimization", AIAA-85-3095, October, 1985.

5. Elias, Antonio L., "Computer-Aided Engineering: The AI Connection", AIAA Astronautics and Aeronautics, July/August 1983, pp. 48-54.

6. Tong, S.S. "Coupling Artificial Intelligence and Numerical Computation for Engineering Design" AIAA-86-0242, AIAA 24th Aerospace Sciences Meeting, January 6-9, 1986.

7. Rajagopolan, R., and Siddiqi, S., "Aircraft Turbojet Engine Modelling Using Artificial Intelligence Techniques", AIAA-85-0253, AIAA 23rd Aerospace Sciences Meeting, January, 1985.

8. Moseley, W., and Zeagler, M., "Embedding Artificial Intelligence in Existing Applications", SAE 860336, International Congress and Exposition, Feb. 1986.

9. Oza, R.D., "Some of the Issues in the Implementation of Artificial Intelligence in Engineering Design", SAE 860340, International Congress and Exposition, Feb., 1986.

10. Joyce, J.D. and Uthurusamy, R., "Promises and Pitfalls of Knowledge Systems: An Overview", SAE 860333, International Congress and Exposition, Feb., 1986.

11. Dankel II, D.D., "Expert Systems: Misconceptions and Reality", SAE 860334, International Congress and Exposition, Feb., 1986.

12. Kutler, P., and Mehta, U., "Computational Aerodynamics and Artificial Intelligence", AIAA P84-1531, AIAA 17th Fluid Dynamics, Plasma Dynamics, and Lasers Conference, June, 1984.

13. Tong, S.S. "Design of Aerodynamic Bodies Using Artificial Intelligence/Expert System Technique", AIAA Paper 85-0112, Reno, Nev., 1985.

14. "Europeans Accelerate Work in Artificial Intelligence", Aviation Week and Space Technology, March 10, 1986, pp. 261-266.

15. Nicklaus, D., Tong, S., and Russo, C., "ENGINEOUS, A Knowledge Directed Computer Aided Design Shell", submitted for presentation at the Third IEEE Conference on Artificial Intelligence Applications", Feb. 22-28, 1987.

16. "KEE", trademark for a software tool marketed by Intellicorp, Menlo Park, California. See Fikes, R.E., and Kehler, T.P., "The Role of Frame-Based Representation in Reasoning" Communication of the ACM on Knowledge Based Systems, Sept. 1985.

17. Sterle, G.L., Jr., "Common LISP, The Language," Digital Press, Burlington, Mass., 1984.

18. Smith, George E., "The Dangers of CAD", Mechanical Engineering, February, 1986, pp 58-64.