ENGINEERING TRANSACTIONS • Engng. Trans. • 44, 3–4, 361–373, 1996
Polish Academy of Sciences • Institute of Fundamental Technological Research

MINIMUM WEIGHT DESIGN OF TRUSSES USING HEURISTIC RULES

M. PYRZ (LILLE)

The minimum weight design of truss structures, composed of elements chosen from catalogues of available profiles, is presented in the paper. A controlled enumeration method is supplied with a knowledge module, containing the problem-oriented information, and represented symbolically. Heuristic rules, proposed on the basis of static analysis of the structure, are used to eliminate all "non-promising" propositions without numerical checking for feasibility. This approach leads to a great reduction (with respect to the "standard" enumeration) in the number of variants that have to be verified to find the global optimum. The application of knowledge-based enumeration is illustrated by numerical examples of standard truss optimisation problems for one or several loading conditions.

1. Introduction

Many engineering problems involve discrete variables. It is not rare that parameters of the real-life structures are based on typical standard components. The characteristics of rolled steel bars the truss structures are made of, have to be chosen from a set of commercially available standard profiles. The optimal design in such practical situations, when the design variables are not continuous-valued, needs the application of adequate methods. The simplest approach is to solve first an equivalent problem in continuous design variable space, and then to round off the solution to the nearest allowable discrete values. The result may be not only non-optimal but, in fact, infeasible. Several computational techniques have been developed so far to handle discrete or integer values in engineering problems. The first IUTAM Symposium on Discrete Structural Optimisation [1] emphasises the considerable attention of researchers in this field. The surveys of different approaches, applied to discrete optimum structural design, are presented for example in [2, 3, 4]. The most important methods can be classified into branch and band, dual, enumeration and penalty function. In addition, in the papers which appeared in recent years, the promising applications of genetic algorithms are explored.

The controlled enumeration methods search for the optimal solution of a discrete optimisation problem by a partial enumeration, without checking all possible combinations. The number of design variable sets, that have to be verified, can be very large in real problems. Verification of constraints for many generated variants is usually connected with expensive numeric processing and a considerable time of calculus. The enumeration algorithm would be more efficient if "non-promising" candidates were eliminated from the checking constraints procedures.

The knowledge of the problem to be solved can often substantially reduce the computational effort and this statement motivates the proposed approach. The Artificial Intelligence techniques have made possible the representation of a knowledge in symbolic terms, and this way, integration and processing of a broader range of information in computer programs. The coupling of the AI and classical numerical procedures in engineering problems emerged in the creation of the so-called Expert Systems. The applications of the knowledge-based approach to the design optimisation follow this trend [5,6,7].

In the presented approach the symbolical and numerical computations are coupled in one computer program to form a knowledge-based optimisation algorithm, joining advantages of the traditional systems of numerical analysis and those of knowledge-based systems. The main idea is to demonstrate the potential of the knowledge applied to engineering optimisation and to show, how an additional information of the problem can be applied to enhance "standard" numerical approach. A controlled enumeration algorithm is supplied with a symbolic processing module, containing the problem-oriented rules. A method for formulation of heuristics for minimum weight truss optimisation is proposed. The information on the problem under consideration is next used to eliminate the useless constraints verification for the propositions considered to be "non-promising", and to remove *a priori* all unrealistic, redundant and infeasible variants without checking them numerically for feasibility. The knowledge base consists of facts, rules and heuristics, expressed in the pseudo-natural language of the task, and accompanied by a reasoning technique of the inference engine. A rule-based production system representation of the Prolog language has been chosen. The effectiveness of the method is illustrated by two standard numerical examples for the minimum weight design of truss structures under one or several loading conditions.

2. Formulation of the optimisation problem

Discrete optimisation characterised by a linear objective function and arbitrary constraints is considered in the presented approach. A generalised mathematical formulation of this problem is as follows:

find $\mathbf{x}$ such that

$$W(\mathbf{x}) = \mathbf{c}\,\mathbf{x} \to \text{minimum} \tag{2.1}$$

subject to:

$$g_k(\mathbf{x}) \leq 0, \quad k = 1, \ldots, K, \tag{2.2}$$

$$h_j(\mathbf{x}) = 0, \quad j = 1, \ldots, J, \tag{2.3}$$

$$\mathbf{x} = [x_1\, x_2 \ldots x_N], \quad x_i \in \{x_i^1;\ x_i^2;\ \ldots;\ x_i^{Mi}\}, \quad i = 1, \ldots, N, \tag{2.4}$$

where $W(\mathbf{x})$ is the linear objective function to be minimised, $\mathbf{x}$ is the vector of N discrete design variables x_i, $\mathbf{c}$ is the vector of N constant, real coefficients c_i characterising the components of the objective function, and $h(\mathbf{x})$, $g(\mathbf{x})$ are arbitrary functions, corresponding respectively to the equality and inequality constraints. The discreteness constraints state that each design variable x_i has to be selected from a corresponding finite set of M_i available discrete values.

In the minimum weight truss optimisation the objective function (2.1) is the weight of the structure, the design variables x_i are cross-sectional areas of bars, and the coefficients c_i correspond to the lengths of bars multiplied by the material density ϱ. The inequalities (2.2) represent the limits imposed on response quantities from the structure analysis such like constraints on displacements, stresses, buckling, etc. The constraints (2.3) correspond to the equilibrium equations. The characteristics of bar elements have to be chosen from finite catalogues of available profiles (2.4).

3. Knowledge-based enumeration approach

The solution of the problem (2.1) – (2.4) is based on a modified version of the enumeration method according to the non-decreasing values of the objective function [8], which leads to the global optimum. The method generates an ordered sequence of design variable vectors

$$\mathbf{x}_1, \quad \mathbf{x}_2, \quad \mathbf{x}_3, \quad \ldots \tag{3.1}$$

according to the non-decreasing values of the corresponding objective functions

$$W_1 \leq W_2 \leq W_3 \leq \ldots, \qquad W_i = W(\mathbf{x}_i). \tag{3.2}$$

The constraints of the problem have to be verified for all sequentially generated vectors $\mathbf{x}_i$, $i = 1, 2, \ldots$. The first design variable set $\mathbf{x}^*$, satisfying all requirements, gives the optimal solution $W_{\text{opt}} = W(\mathbf{x}^*)$ of the minimisation problem. The algorithm can be started from any value stated as a lower bound for the objective function values W_i to be generated in non-decreasing order.

The optimisation procedure would be more efficient if one eliminated "non-promising" candidates without checking them numerically for feasibility. A lot of design variable sets can be removed *a priori* thanks to the character of the expected results and the knowledge of the problem to be solved. New styles of programming based on symbolic representation (like AI) allow us to represent much more and significantly different engineering knowledge.

The proposed knowledge-based enumeration algorithm is composed of three separate modules, corresponding to different levels of processing (Fig. 1).

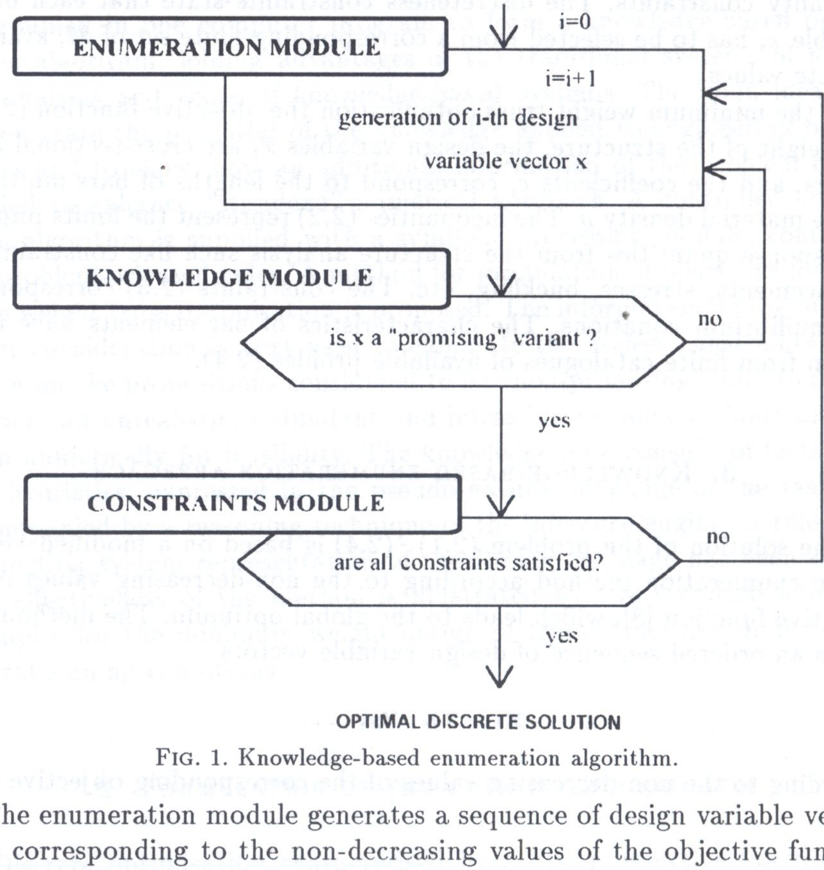

FIG. 1. Knowledge-based enumeration algorithm.

The enumeration module generates a sequence of design variable vectors (3.1) corresponding to the non-decreasing values of the objective function (3.2). For these vectors the constraints of the problem have to be verified.

The knowledge module acts as a filter between the generation of candidates and the verification of the constraints to limit the explored discrete design space. The information on the problem is expressed symbolically in the pseudo-natural language of the task, using the first order predicate logic of the Prolog language. The knowledge base contains specific heuristic rules that are used to remove the candidates from the checking constraints procedure if they are considered to be unrealistic *a priori*. The "IF condition THEN action" rule-based production system representation has been chosen. It enables us to stay close to the language used by designers, and this way, to exploit easily the descriptive knowledge of a particular problem. The infeasible propositions are skipped and, as a result, a sequence of "promising" variants corresponding to the non-decreasing values of the objective function is obtained.

Finally, the constraints module checks for feasibility the design variable variants coming from the knowledge module. The first design vector satisfying the constraints (2.2), (2.3) and (2.4) of the problem is the optimal solution.

4. Formulation of heuristic rules for truss structures

The symbolically coded information on the particular problem to be solved can complete the conventional numerical algorithm. It can be obtained from the analysis of the structure, manufacturing or technological constraints, designer's experience etc. A detailed description of the typology of engineering knowledge, that can be integrated into AI based computing, can be found in [9]. In the case of truss structures, the mechanical behaviour analysis can be used to obtain heuristic rules. The following simple procedure, based on the static analysis, enables the formulation of relations between the design variables. It is the author's conclusion from the first attempts of introducing AI techniques to optimal sizing of trusses [10, 11]. The proposed approach is not intended to be of a general nature, but it can be successfully applied in the case of statically loaded truss structures. It is presented here to illustrate the importance and the potential of an additional information which can considerably enhance the standard optimisation approach.

The description presented below corresponds to the case of structures subjected to multiple independent static load conditions. If only one loading case is taken into account, the last step of the algorithm must be neglected. If several bars of a truss are assembled into linking zones, grouping elements of the same cross-sectional characteristics, the corresponding average values in linking zones have to be used.

Separately for every load condition and for equal values of all design variables:

- determine the member stresses and the average stresses in all bars (zones);
- formulate a "high certainty" relations of type "greater than" or "less than" between element stresses in bars (zones);
- translate the obtained expressions to "design variable language" using the hypothesis, that "at the optimum, the more stressed elements will correspond to greater cross-sectional areas than those of less stressed bars";
- finally, if the optimisation problem invokes multiple load conditions, find the common part of all the obtained relations, corresponding to the non-contradictory statements.

A certainty coefficient, denoted by ψ, has to be applied to obtain a "safe" evaluation of the compared element stresses. The word "greater than" has to mean here "greater than at least ψ per cent of the value in question", when for example $\psi = 300\%$.

The expressions obtained in this way can be viewed as a predimensioning of cross-sectional areas of bars. The presizing estimation is based on a uniform stress distribution supposed at the optimum. The coefficient ψ can be interpreted as a representation of an intuitive human factor in the processes of the formulation of rules.

5. Numerical examples

Two standard examples of truss optimisation are analysed for the modulus of elasticity $E = 10^7$ psi and the density $\varrho = 0.1\,\mathrm{lbs/in}^3$ corresponding to aluminium. The Anglo-Saxon units have been kept for the reason of compatibility with the results known from literature. The calculus has been carried out for static elastic linear approach. To improve the effectiveness of the enumeration search, the starting value for the enumeration algorithm was fixed to the continuous optimal solution, taken from the referenced papers.

The number of candidates that have to be checked for feasibility to find the discrete optimum is compared for the knowledge-based approach and the "standard" enumeration (without any problem-oriented information included). The difference between the two methods is a measure of the effectiveness of the applied heuristic rules.

5.1. Ten-bar truss

The weight minimisation of the classical ten-bar truss structure (Fig. 2) is considered. The structure is designed to support two loads $P = 10^5$ lb applied in nodes 2 and 4, acting in downward direction. The optimal results from [12] have been taken as reference values. The structure is optimised subject to the maximum displacement limit of ±2.0 in for all nodes in the X and Y directions, and to the maximum elastic stress limit $\sigma_{\max} = \pm 25000$ psi in all members. The design variables are cross-sectional areas A_i of 10 bars ($i = 1, \ldots, 10$) from the following catalogues of discrete sections:

$$A_1,\ A_3,\ A_4,\ A_8,\ A_9 \in \{12.0;\ 15.0;\ 18.0;\ 20.0;\ 25.0;\ 30.0;\ 35.0;\ 45.0\}\ [\text{in}^2],$$
$$A_2,\ A_5,\ A_6,\ A_7,\ A_{10} \in \{0.1;\ 1.0;\ 2.0;\ 5.0;\ 8.0;\ 12.0;\ 15.0;\ 18.0\}\ [\text{in}^2].$$

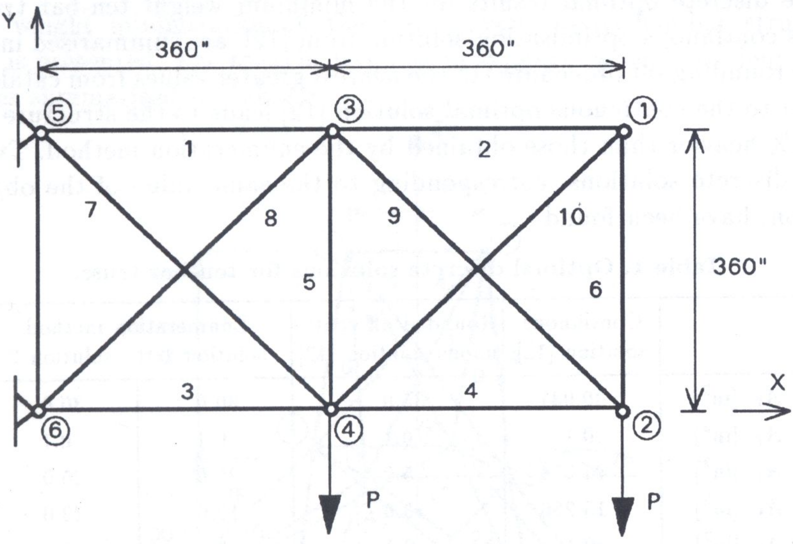

FIG. 2. Ten-bar truss.

The static analysis of the truss for equal values of all cross-sectional areas and for the certainty coefficient $\psi = 300\%$, has given the following relations between stresses

$$\sigma_1 > \sigma_2\,,\quad \sigma_1 > \sigma_4\,,\quad \sigma_1 > \sigma_5\,,\quad \sigma_1 > \sigma_6\,,\quad \sigma_1 > \sigma_{10}\,,$$
$$\sigma_3 > \sigma_2\,,\quad \sigma_3 > \sigma_4\,,\quad \sigma_3 > \sigma_5\,,\quad \sigma_3 > \sigma_6\,,\quad \sigma_3 > \sigma_{10}\,,$$
$$\sigma_7 > \sigma_2\,,\quad \sigma_7 > \sigma_5\,,\quad \sigma_7 > \sigma_6\,,\quad \sigma_7 > \sigma_{10}\,,$$
$$\sigma_8 > \sigma_2\,,\quad \sigma_8 > \sigma_5\,,\quad \sigma_8 > \sigma_6\,,\quad \sigma_8 > \sigma_{10}\,.$$

They enabled the formulation of five simple heuristics that are presented below in the form as they look like in the computer program.

IF *10_bar_truss* and *2_loads_in_nodes_2_and_4*
THEN *A1_more_stressed* and *A3_more_stressed*
and *A7_more_stressed* and *A8_more_stressed.*

IF *A1_more_stressed*
THEN $A_1 > A_2$ and $A_1 > A_4$ and $A_1 > A_5$ and $A_1 > A_6$ and $A_1 > A_{10}$.
IF *A3_more_stressed*
THEN $A_3 > A_2$ and $A_3 > A_4$ and $A_3 > A_5$ and $A_3 > A_6$ and $A_3 > A_{10}$.
IF *A7_more_stressed*
THEN $A_7 > A_2$ and $A_7 > A_5$ and $A_7 > A_6$ and $A_7 > A_{10}$.
IF *A8_more_stressed*
THEN $A_8 > A_2$ and $A_8 > A_5$ and $A_8 > A_6$ and $A_8 > A_{10}$.

The discrete optimal results for the minimum weight ten-bar truss, as well as continuous optimisation solution from [12], are summarised in Table 1. The rounding-off procedure (to the nearest greater values from catalogues) applied to the continuous optimal solution [12] leads to the structure which is 12.6% heavier than those obtained by the enumeration method. Two different discrete solutions, corresponding to the same value of the objective function, have been found.

Table 1. Optimal discrete solutions for ten-bar truss.

	Continuous solution [12]	Rounded off continuous solution [12]	Enumeration method solution 1	Enumeration method solution 2
A_1 [in^2]	30.031	35.0	30.0	30.0
A_2 [in^2]	0.1	0.1	0.1	0.1
A_3 [in^2]	23.274	25.0	25.0	25.0
A_4 [in^2]	15.286	18.0	12.0	12.0
A_5 [in^2]	0.1	0.1	0.1	0.1
A_6 [in^2]	0.5565	1.0	1.0	1.0
A_7 [in^2]	7.4683	8.0	8.0	8.0
A_8 [in^2]	21.618	25.0	20.0	25.0
A_9 [in^2]	21.618	25.0	25.0	20.0
A_{10} [in^2]	0.1	0.1	0.1	0.1
Weight [lbs]	5061.6	5809.169	5158.6106	5158.6106

In Table 2 the number of variants that have to be checked to find the discrete optimum is compared for the knowledge-based and "standard" enumeration (without the knowledge module). The number of verified variants for "standard" enumeration was chosen as a reference value (100%). It is seen from the presented example that the knowledge-based approach implies an enormous reduction (with respect to the "standard" version) in the number of variants that have to be checked to find the discrete optimum. For a simple knowledge base composed of five rules, only for 8.49% of the generated variants, the constraints had to be verified to reach the optimum.

Table 2. Number of checked variants necessary to find the discrete optimum for 10-bar truss.

Standard enumeration method (no knowledge base)	100%	12 691 604
Knowledge-based enumeration method	8.49%	1 076 912

5.2. Twenty-five-bar truss under two loading conditions

The weight minimisation of the well known twenty-five-bar structure (Fig. 3) is presented. The truss is designed to support two independent loading cases summarised in Table 3.

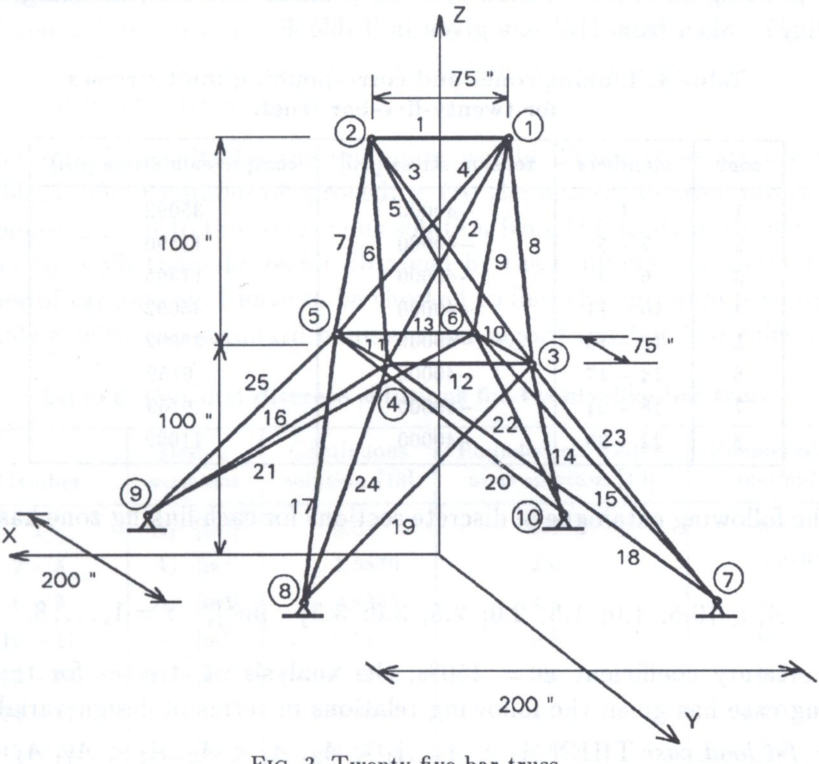

FIG. 3. Twenty-five-bar truss.

The structure is optimised under the constraints imposed on allowable maximal displacement ±0.35 in for all nodes in the X, Y, Z directions,

Table 3. Loading components for twenty-five-bar truss.

Loading case	Node	Load X [lbs]	Load Y [lbs]	Load Z [lbs]
1	1	1000	10000	−5000
	2	0	10000	−5000
	3	500	0	0
	6	500	0	0
2	1	0	20000	−5000
	2	0	−20000	−5000

and stresses and buckling limits in all members. The elastic stress limit is $\sigma_{\max} = \pm 40000$ psi. The 25-member elements are linked to 8 variables, the cross-sectional areas of bars in 8 zones of identical elements, maintaining symmetry of the structure. The optimal continuous results from [13] have been taken as reference values. The definition of linking zones as well as the corresponding allowable tension and compression stresses (including elastic buckling), taken from [13], are given in Table 4.

Table 4. Linking zones and corresponding limit stresses for twenty-five-bar truss.

zone	members	tension stress [psi]	compression stress [psi]
1	1	−40000	35092
2	2 − 5	−40000	11590
3	6 − 9	−40000	17305
4	10 − 11	−40000	35092
5	12 − 13	−40000	35092
6	14 − 17	−40000	6759
7	18 − 21	−40000	6759
8	22 − 25	−40000	11082

The following catalogue of discrete sections for each linking zone has been chosen:

$$A_i \in \{0.5;\ 1.0;\ 1.5;\ 2.0;\ 2.5;\ 3.0;\ 3.5\} \quad [\text{in}^2], \quad i = 1, \ldots, 8.$$

For certainty coefficient $\psi = 450\%$, the analysis of stresses for the first loading case has given the following relations in terms of design variables:

IF *1st_load_case* THEN $A_1 < A_2$, $A_1 < A_3$, $A_1 < A_6$, $A_1 < A_7$, $A_1 < A_8$,
$A_4 < A_2$, $A_4 < A_3$, $A_4 < A_6$, $A_4 < A_7$, $A_4 < A_8$,
$A_5 < A_2$, $A_5 < A_3$, $A_5 < A_7$, $A_5 < A_8$,
$A_6 < A_3$, $A_6 < A_8$.

The same analysis for the second loading case has led to

IF *2nd_st_load_case* THEN $A_1 < A_2, A_1 < A_3, A_1 < A_7,$
$A_4 < A_2, A_4 < A_3, A_4 < A_7,$
$A_5 < A_2, A_5 < A_3, A_5 < A_7,$
$A_6 < A_2, A_6 < A_3, A_6 < A_7,$
$A_8 < A_2, A_8 < A_3, A_8 < A_7.$

The common part of the two sets results in expressions, precising simply that the cross-sectional areas of certain bars have to be greater than those of the other bars, according to the supposed stress distribution at the optimum. The heuristics are included into the knowledge base and they are presented below in the form as they look like in the computer program.

IF *25_bar_truss* and *2_load_conditions*
THEN *zone1_less_stressed* and *zone4_less_stressed*
and *zone5_less_stressed* and *condition1*.
IF *zone1_less_stressed* THEN $A_1 < A_2$ and $A_1 < A_3$ and $A_1 < A_7$.
IF *zone4_less_stressed* THEN $A_4 < A_2$ and $A_4 < A_3$ and $A_4 < A_7$.
IF *zone5_less_stressed* THEN $A_5 < A_2$ and $A_5 < A_3$ and $A_5 < A_7$.
IF *condition1* THEN $A_6 < A_3$.

The optimal results for the minimum weight 25-bar truss are presented in Table 5. The rounding-off procedure (to the nearest greater values from catalogue) applied to the continuous solution from [13] leads to the structure heavier by 8.8% than the result obtained by the enumeration method. The number of variants that have to be checked to find the optimum is compared in Table 6 with the standard enumeration algorithm. For five rules in the

Table 5. Optimal discrete solutions for twenty-five-bar truss.

Member	Design variables	Continuous solution [13]	Rounded off conti– nous solution [13]	Enumeration method
1	A_1 [in^2]	0.01	0.5	0.5
2 – 5	A_2 [in^2]	1.9870	2.0	2.0
6 – 9	A_3 [in^2]	2.9935	3.0	3.5
10 – 11	A_4 [in^2]	0.01	0.5	0.5
12 – 13	A_5 [in^2]	0.01	0.5	0.5
14 – 17	A_6 [in^2]	0.6840	1.0	0.5
18 – 21	A_7 [in^2]	1.6769	2.0	1.5
22 – 25	A_8 [in^2]	2.6621	3.0	3.0
Weight [lbs]		545.16271	628.8399	577.74305

knowledge base only for 3.14% of the generated variants, the constraints had to be verified to reach the optimum compared with the "standard" enumeration version.

Table 6. Number of checked variants necessary to find the discrete optimum for 25-bar truss.

Standard enumeration method (no knowledge base)	100%	419 395
Knowledge-based enumeration method	3.14%	28 714

6. Concluding remarks

Including non-algorithmical and non-numerical ability into "conventional" programs can improve performances of the engineering-oriented optimisation tools. The potential of symbolic computations applied to the problems of discrete optimisation has been emphasised.

The knowledge-based approach can imply a considerable reduction (with respect to the "standard" version) in the number of variants that have to be checked to find the global discrete optimum. The simple heuristics, formulated in the paper rather for illustrative purposes, enable a great decrease in the dimension of the design space explored numerically by the constraints module of the enumeration method. For small knowledge base, the number of numerically checked variants is proportional to the CPU time needed for the verification of the constraints.

A domain-specific knowledge is an active component of the discrete optimisation algorithm. The knowledge module applies the information to eliminate "incorrect" design variables sets. The rules are represented in a "natural language" of the problem, and are easy to change or modify. The algorithm does not require much computer memory and can be easily adapted to parallel processing.

In the presented numerical examples, the decisions of skipping non-promising variants have been obtained from the mechanical behaviour analysis. Other sources of information like technology, economical properties, designer's experience or utilities aspects can be used as well to achieve a high level of expert knowledge in engineering optimisation problems. It is hoped that the knowledge-based approach used in conjunction with numerical techniques can considerably improve the performances of other conventional optimisation procedures.

References

1. W. Gutkowski and J. Bauer [Eds.], *Discrete structural optimisation*, IUTAM Symposium Zakopane, Poland, 31.08-03.09.1993, Springer Verlag, 1994.
2. J. Bauer, *A survey of methods for discrete optimum structural design*, CAMES, **1**, pp. 27–38, 1994.
3. J.S. Arora, M.W. Huang and C.C. Hsieh, *Methods for optimisation of nonlinear problems with discrete variables. A review*, Struct. Optim., **8**, pp. 69–85, 1994.
4. P.B. Thanedar and G.N. Vanderplaats, *Survey of discrete optimisation for structural design*, J. Struct. Engng., **121**, 2, pp. 301–306, 1995.
5. M. Balachandran, *Knowledge-based optimum design*, [in:] Topics in Engineering, vol. 10, C.A. Brebbia, J.J. Connor [Eds.], Computational Mechanics Publications, Southampton UK, Boston USA 1993.
6. M. Kleiber [Ed.], *Artificial intelligence in computational engineering*, Ellis Horwood Limited, 1990.
7. M. Balachandran and J. Gero, *Knowledge engineering and multicriteria optimization*, [in:] Multicriteria Design Optimisation, Procedures and Applications, H. Eschenauer, J. Koski, A. Osyczka [Eds.], pp. 115–147, Springer Verlag, Berlin, Heideberg, New York 1990.
8. Z. Iwanow, *An algorithm for finding an ordered sequence of a discrete linear function*, Control and Cybernetics, **19**, 3-4, pp. 129–154, 1990.
9. C.L. Dym and R.E. Levitt, *Towards the integration of knowledge for engineering modeling and computation*, Engng. with Computers, **7**, pp. 209–224, 1991.
10. M. Pyrz, *Symbolic computations approach in controlled enumeration methods applied to discrete optimisation*, [in:] Discrete Structural Optimisation, IUTAM Symposium Zakopane, Poland, 31.08-03.09.1993, W. Gutkowski, J. Bauer [Eds.], pp. 221-227, Springer Verlag, 1994.
11. M. Pyrz, *Knowledge-based discrete optimisation of truss structures*, Engng. Trans., **47**, 1-2, pp. 313–325, 1995.
12. E.J. Haug and J.S. Arora, *Applied optimal design. Mechanical and structural systems*, J. Wiley, New York 1979.
13. G.I.N. Rozvany and M. Zhou, *New discretized optimality criteria method. State of the art*, [in:] Discrete Structural Optimisation, IUTAM Symposium Zakopane, Poland, 31.08-03.09.1993, W. Gutkowski, J. Bauer [Eds.], pp. 118–134, Springer Verlag, 1993.

ECOLE UNIVERSITAIRE D'INGENIEURS DE LILLE
LABORATOIRE DE MECANIQUE DE LILLE, LILLE, FRANCE.

Received April 10, 1996.