*ENGINEERING TRANSACTIONS* • *Engng. Trans.* • **43**, 1–2, 313–325, 1995
Polish Academy of Sciences • Institute of Fundamental Technological Research

# KNOWLEDGE-BASED DISCRETE OPTIMIZATION OF TRUSS STRUCTURES (*)

M. PYRZ (LILLE)

The knowledge-based approach to discrete optimization is presented in the paper. The minimization problem characterized by linear objective function and arbitrary constraints is considered when design variables have to be chosen from a set of discrete values available. The controlled enumeration algorithm according to the non-decreasing values of the objective function is supplied with an additional module manipulating the information represented symbolically. This module contains the domain-oriented knowledge expressed in the form of heuristic rules and is used to eliminate the useless constraints verification for the propositions considered to be "non-promising". The approach coupling the symbolic and numerical computations enables a significant reduction in the number of design variables variants that must be checked for feasibility in order to find the optimum. The numerical examples for the minimum weight optimization of a cantilever truss structure and the corresponding simple heuristic rules are presented.

## 1. Introduction

The modern engineering design is based in many cases on prefabricated components chosen from commercially available standard elements. It is not rare that some parameters characterizing the number, positions or interrelations between structural components can take only integer or discrete values. The structures to be composed using rolled beams or metal sheets from catalogues, the systems with constraints imposed on nodes or support location are examples of such situations. This type of practical applications involves variables which are not continuous.

The solution of optimization problems in which some of (or all) design variables are discrete and should be chosen from a set of available values needs the application of adequate methods. The discrete variable engineering optimization dates from early 1970s and several techniques have been developed so far. The recent survey of different approaches to the discrete

(*) Paper presented at 30th Polish Solid Mechanics Conference, Zakopane, September 5–9, 1994.

optimum structural design can be found in [1], where the methods have been classified into branch and band, dual, enumeration, penalty function, simulated annealing and others. The first IUTAM Symposium on Discrete Structural Optimization [2], held in Poland in 1993, emphasizes the considerable attention paid by the researchers to this field.

The idea of controlled enumeration methods applied to the discrete optimization consists in using such algorithms which give the optimal solution by a partial enumeration, without checking all the feasible variants. The combinatorial techniques are applied to generate a sequence of design variable vectors for which the constraints of the problem are checked until the criteria for the solution are satisfied. The number of design variable sets that have to be verified to find the global optimum can be very significant in real problems due to the combinatorial explosion of possible propositions. The constraints checking is usually connected with a heavy numeric processing and implies a considerable time of calculus.

The knowledge of the problem to be solved can often substantially reduce the computational effort. The enumeration technique would be more efficient if "non-promising" candidates were eliminated from the constraints checking procedure. The information about the problem under consideration can be applied to remove the *a priori* unrealistic, redundant and infeasible variants. The Artificial Intelligence techniques enable powerful processing of symbolic data and are successfully applied in engineering problems. The knowledge-based approach to the design optimization follows this trend (numerous papers in [3, 4, 5]).

The purpose of this paper is to present a knowledge-based approach to the discrete optimization of engineering structures using a controlled enumeration algorithm. The symbolic and numerical computations are coupled in one computer program to form a knowledge-based optimization algorithm, joining advantages of the traditional systems of numerical analysis and those of knowledge-based systems. The enumeration algorithm according to the nondecreasing values of the objective function is supplied with an additional module manipulating the information represented symbolically. This module contains the domain-oriented knowledge expressed in the form of heuristic rules and is used to eliminate the useless constraints verification for the propositions considered to be "non-promising". The coupling symbolic and numerical computations approach enables a significant reduction (with respect to the "standard enumeration") in the number of design variables variants that have to be checked for feasibility to find the optimum. The effectiveness of the method is illustrated by the minimum weight optimization of a cantilever truss structure. A decrease in the numerical effort necessary to find the global minimum, observed in all examples, demonstrates the potential of the proposed approach applied to engineering optimization problems.

## 2. Problem statement

The discrete optimization of the problems characterized by a linear objective function and arbitrary constraints is considered in the paper.

The minimization problem can be generally formulated for $N$ discrete design variables $x_i$ $(i = 1, \ldots, N)$ as follows:

minimize

$$f(\mathbf{x}) = \sum_{i=1}^{N} c_i x_i = \mathbf{c}\mathbf{x}, \tag{2.1}$$

subject to arbitrary constraints

$$\mathbf{x} \in S, \tag{2.2}$$

where $f(\mathbf{x})$ is the linear objective function to be minimized, $\mathbf{x}$ is the vector of $N$ discrete design variables $x_i$, $\mathbf{c}$ is the vector of $N$ constant real coefficients $c_i$ characterizing the components of the objective function, and $S$ stands for a set of feasible solutions determined by arbitrary constraints.

The discreteness constraints state that each design variable $x_i$ $(i = 1, \ldots, N)$ has to be selected from a finite set $D_i$ of $m_i$ feasible discrete values

$$x_i \in D_i = \{x_1^i, x_2^i, x_3^i, \ldots, x_{m_i}^i\}, \qquad i = 1, \ldots, N. \tag{2.3}$$

The solution of the problem under consideration is based on a combinatorial algorithm of controlled enumeration. The enumeration method according to the non-decreasing values of the objective function [6] has been chosen. It guarantees the global optimum solution. Its effectiveness is, so far, limited to smaller problems due to a combinatorial explosion of possible variants. A knowledge of the problem to be solved is included in the solution algorithm and substantially reduces the computational effort. The main ideas of the method have been already presented by the author in [7]. The presented approach can be applied to a larger class of discrete problems using some linearisation methods for formulation of the nonlinear objective function.

## 3. Knowledge of the problem to be solved

The traditional optimization programs are concentrated on the numerical aspects of the solution of the problem. They use efficiently and precisely the procedural methods and are rather of a "black box" nature. The information encoded implicitly in the algorithms is generally determined by the expected behaviour of the simulated domain. The design process involves many heuristic aspects, changing from one task to another. They can be difficult to incorporate into a traditional computer program, even if they are accessible in an explicit form.

Advances in the AI techniques have led to the knowledge-based systems which provide tools for explicit representation and efficient processing of knowledge in some domains. The specific, problem-oriented knowledge can complete the information contained in the conventional optimization algorithm to form a more efficient tool for engineering optimization.

The domain-oriented information useful in the optimization of engineering problems can result from:

- mechanical behaviour of the structure or given manufacturing, technological or economical constraints;
- methodology of modelling for the considered class of structures;
- designer's experience based on previous results calculated for similar problems;
- decision-making process for "equivalent" or "competitive" propositions of solutions;
- utilities aspect, functional specifications, simplicity, aesthetic functions, visual characteristics, etc.

The main idea of the approach proposed is to couple numerical and symbolic processing in one algorithm. The information about the problem to be optimized is expressed in the pseudo-natural language of the task using the Prolog language. The knowledge base consists of domain facts, rules and heuristics associated with the problem and accompanied by a reasoning technique. An "IF condition THEN action" rule-based production system representation has been chosen. It is easy to change or actualise thanks to the declarative characteristics of the representation: one should only decide what the system has to know.

## 4. Method of solution

The following sections present two approaches to the solution of the problem in question. The first one, called "standard enumeration", is a classical approach based on a controlled enumeration algorithm. The second is the knowledge-based modification of the first one. In the numerical examples, the optimal results obtained by both methods are compared. The number of design variables variants that have to be checked for feasibility to find the global optimum is studied to analyse the effectiveness of the proposed heuristic rules.

### 4.1. "Standard enumeration" method

In the paper a version of the enumeration method according to the non-decreasing values of the objective function [6] has been adapted to the discrete optimization problem (2.1)–(2.3).

The idea of the method is to find an ordered sequence of design vectors

$$\mathbf{x}_1;\ \mathbf{x}_2;\ \mathbf{x}_3,\ \ldots \tag{4.1}$$

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

$$f_1 \le f_2 \le f_3 \le \ldots, \qquad f_i = f(\mathbf{x}_i). \tag{4.2}$$

If min is the smallest natural $n \in N$, for which the condition $\mathbf{x} \in S$ is satisfied, then the solution of the minimization problem is

$$\mathbf{x}_{\mathrm{opt}} = \mathbf{x}_{\mathrm{min}}, \qquad f_{\mathrm{opt}} = f(\mathbf{x}_{\mathrm{opt}}). \tag{4.3}$$

For the ordered coefficients $c_i$ of the objective function, the algorithm constructs recursively a virtual tree structure, assigning a unique value of $f_i$ and $\mathbf{x}_i$ to each vertex. The suitable vertices (or all groups of vertices) are next examined making use of the monocity properties of the created graph. All variants of $\mathbf{x}_i$ corresponding to the same value of the objective function are found at each step. The algorithm guarantees the global minimum solution. The search can be started from any arbitrary value, stated as a lower bound for the objective function values to be generated in non-decreasing order. It means, that the search would be accelerated, if the solution of the equivalent problem formulated for continuous design variables was already known. The algorithm does not require much computer memory. This approach is schematically represented in the Fig. 1. In reality, the design variable vectors are generated in packets corresponding to the same value of the objective function.

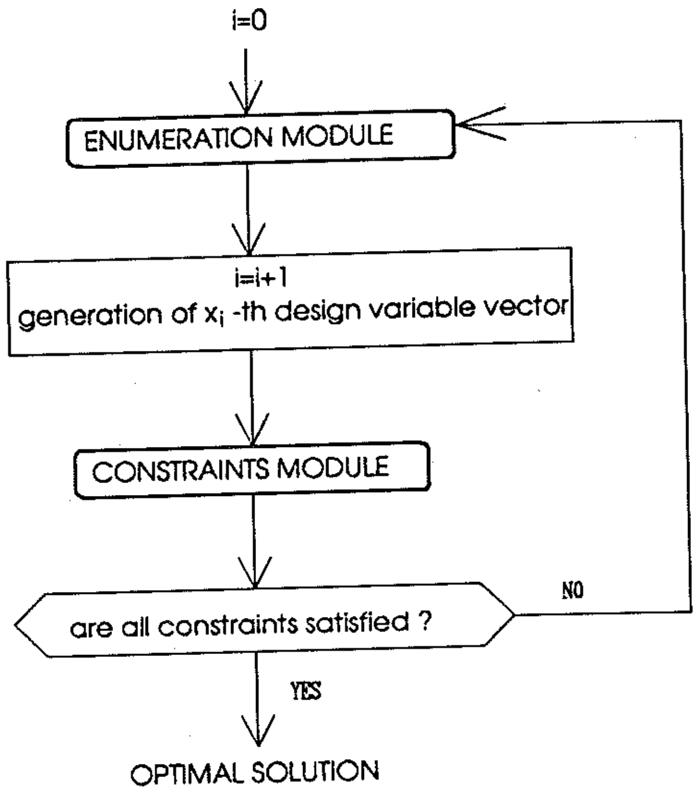


FIG. 1. Algorithm of the "standard enumeration" method.

### 4.2. *Knowledge-based enumeration*

The enumeration method presented in the previous chapter needs verification of the constraints for all subsequently generated design variables vectors $\mathbf{x}_1$ $\mathbf{x}_2$ $\mathbf{x}_3$ ... until reaching a $\mathbf{x}_{\min}$ satisfying (2.2). The optimization 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 an understanding of the expected results and a knowledge of the problem to be solved. Many decisions of skipping non-promising variants can result from the technology, manufacturing, mechanical or other problem-oriented properties that can be stated "without calculus". The different forms of an incorporated, "domain-oriented", knowledge can improve the efficiency of the optimization process.

The main idea of the proposed knowledge-based approach is to create three separate modules corresponding to different levels of processing (Fig. 2). The explicit separation of modules is a natural consequence of the application of a controlled enumeration method.

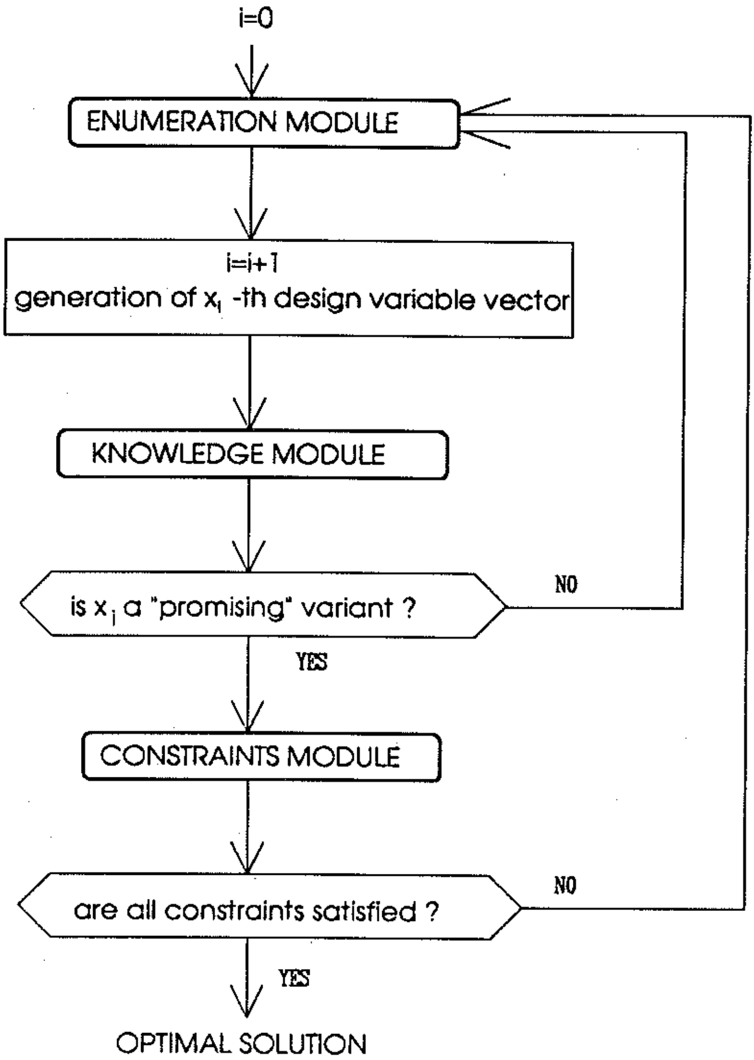


FIG. 2. Algorithm of the "knowledge-based enumeration" method.

*4.2.1. Enumeration module.* The enumeration module generates a sequence of design variable vectors (4.1) corresponding to the non-decreasing values of the objective function (4.2). For these vectors the constraints of the problem have to be verified. Only the coefficients $c_i$ and the discreteness conditions of type (2.3), defining sets of available discrete values, are needed at this stage.

*4.2.2. Knowledge module.* It manipulates a knowledge symbolically represented in the Prolog and acts as a filter between the generation of candidates and the verification of the constraints. This module removes the candidates from the checking constraints procedure if they are considered infeasible on the basis of the "problem-oriented" information contained in the knowledge base. The symbolic processing module is created to limit "without calculus" a combinatorial explosion of possible variants that have to be verified to find the optimal solution. It contains a specific knowledge of the problem to be solved, however some general rules or heuristics can be included. As a result, a sequence of "promising" variants corresponding to the non-decreasing values of the objective function is generated.

*4.2.3. Constraints module.* In this module the design variable variants issued from the knowledge module are subsequently checked for feasibility. The first design vector satisfying the constraints (2.2) of the problem gives the optimal solution (4.3).

## 5. Numerical examples

### *5.1. Minimum weight design of 18-bar truss structure*

The weight minimization of an 18-bar cantilever truss structure (Fig. 3) is presented to illustrate the proposed approach. The design variables are cross-sectional areas $A_i$ of 18 bars, limited to the case of tubular sections with the thickness $t = 0.2R$, the cross-sectional area $A = 0.36\pi R^2$ and the moment of inertia $I = 0.3625A^2$, where $R$ is the radius. The material characteristics $E = 7 \cdot 10^7\,\mathrm{kN/m^2}$ and $\rho = 27.5\,\mathrm{kN/m^3}$ correspond to aluminium. The structure is loaded by 5 vertical forces $P = 10\,\mathrm{kN}$ acting in the downward direction.

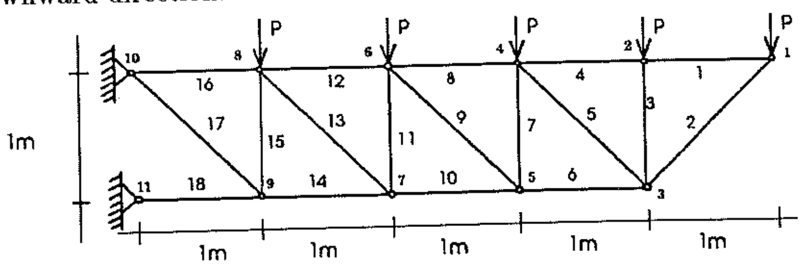


FIG. 3. The 18-bar truss structure.

The structure is optimized for different combinations of constraints imposed on:

vertical displacements $u_j$ of nodes

$$|u_j| \le u_{\max} = 0.05\,\mathrm{m}, \qquad j = 1, 2, ..., 9;$$

elastic stresses

$$|\sigma_i| \leq \sigma_{\max} = 2 \cdot 10^5\,\mathrm{kN/m^2}, \qquad i = 1, ..., 18;$$

buckling stresses

$$\sigma_i \leq \sigma_{\mathrm{buckl}} = 3.5779 E A_i / L_i^2, \qquad i = 1, ..., 18.$$

The calculus have been carried out for geometrically nonlinear formulation.

The optimal solutions are given for two different ways (called Case 1 and Case 2) of dividing the structure into zones of identical elements.

### *5.2. Optimization of 18-bar truss – Case 1*

The structure has been divided into 6 following linking zones, grouping elements of the same cross-sectional characteristics: zone 1 – bars 1, 4, 8; zone 2 – bars 12, 16; zone 3 – bars 6, 10; zone 4 – bars 14, 18; zone 5 – bars 3, 7, 11, 15; zone 6 – bars 2, 5, 9, 13, 17.

The catalogues for discrete sections [in $\mathrm{m}^2 \cdot 0.0001$] in different zones are given below:

$$A_1 \in \{1.0;\ 2.0;\ 3.0;\ 4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0\},$$

$$A_2 \in \{4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0;\ 11.0;\ 12.0;\ 13.0;\ 14.0;\ 15.0\},$$

$$A_3 \in \{1.0;\ 2.0;\ 3.0;\ 4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0\},$$

$$A_4 \in \{4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0;\ 11.0;\ 12.0;\ 13.0;\ 14.0;\ 15.0\},$$

$$A_5 \in \{1.0;\ 2.0;\ 3.0;\ 4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0\},$$

$$A_6 \in \{1.0;\ 2.0;\ 3.0;\ 4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0\}.$$

The full survey of all possible variants would need checking of 1440000 combinations. The starting value for the enumeration algorithm was fixed to 187 N.

The following heuristic rules have been included into the symbolic module of the knowledge-based algorithm. They are formulated on the basis of the statics of the structure and the constraints imposed on displacements. They can be viewed as a predimensioning of the cross-sectional areas of the bars. The rules take into account positions of the loads and tend to prevent the displacement limit excess. The heuristic rules precise the cross-sectional interrelations and state simply, that the sections of the bars supposed to be more stressed are to be greater than those of the less stressed bars. They are presented using the Prolog syntax in the form as they look like in the computer program.

rule 1: IF *downward_acting_loads*
THEN *flexion_of_the_structure*
and *lower_part_compressed.*

rule 2: IF *flexion_of_the_structure*
THEN *left_part_more_stressed.*

rule 3: IF *left_part_more_stressed*
THEN $(A_2 > A_1$ and $A_2 > A_5$ and $A_2 > A_6)$
and $(A_4 > A_3$ and $A_4 > A_5$ and $A_4 > A_6)$
and $(A_2 > A_3)$ and $(A_3 > A_5$ and $A_3 > A_6)$.

rule 4: IF *lower_part_compressed*
THEN $(A_3 > A_1$ and $A_4 > A_1)$.

The results of the optimization are given in Table 1. The number of variants that have to be checked to find the optimum for different constraints is compared for the "standard enumeration" and the "knowledge-based enumeration". The number of equivalent solutions is given but only the first optimal variant is referenced. The best solution among the equivalent ones could be obtained using additional information included in the knowledge module.

**Table 1. Optimal solutions for Case 1.**

| constraints | stress | stress displ. | stress buckling | displ.stress buckling |
|---|---|---|---|---|
| Optimum weight [N] | 226.2817 | 306.0317 | 253.7817 | 317.0317 |
| cross-sectional areas | | * | | ** |
| $A_1$ [m$^2$] | 0.0002 | 0.0003 | 0.0003 | 0.0003 |
| $A_2$ [m$^2$] | 0.0006 | 0.0011 | 0.0006 | 0.0008 |
| $A_3$ [m$^2$] | 0.0004 | 0.0007 | 0.0005 | 0.0006 |
| $A_4$ [m$^2$] | 0.0008 | 0.0013 | 0.0008 | 0.0015 |
| $A_5$ [m$^2$] | 0.0003 | 0.0003 | 0.0005 | 0.0005 |
| $A_6$ [m$^2$] | 0.0004 | 0.0004 | 0.0004 | 0.0004 |
| Number of checked variants for standard enumeration | 62008 | 419492 | 148292 | 490481 |
| Number of checked variants for knowledge-based enumeration | 8507 | 48431 | 18941 | 55620 |

* 5 equivalent solutions exist, ** 10 equivalent solutions exist

### *5.3. Optimization of 18-bar truss – Case 2*

This time the structure has been divided into 6 zones grouping elements of identical cross-sectional characteristics: zone 1 – bars 1, 2, 3, 4; zone 2 – bars 5, 6, 7, 8; zone 3 – bars 9, 10, 11, 12; zone 4 – bars 13, 15, 17; zone 5 – bars 14, 16; zone 6 – bar 18. The linking groups have been chosen to divide the structure into the zones of expected similar stresses.

The catalogues for discrete sections in different linking zones are [in $m^2 \cdot 0.0001$]:

$$A_1 \in \{1.0;\ 2.0;\ 3.0;\ 4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0\},$$
$$A_2 \in \{2.0;\ 3.0;\ 4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0;\ 11.0\},$$
$$A_3 \in \{3.0;\ 4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0;\ 11.0;\ 12.0;\ 13.0;\ 14.0\},$$
$$A_4 \in \{3.0;\ 4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0;\ 11.0;\ 12.0;\ 13.0;\ 14.0\},$$
$$A_5 \in \{4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0;\ 11.0;\ 12.0;\ 13.0;\ 14.0;\ 15.0\},$$
$$A_6 \in \{4.0;\ 5.0;\ 6.0;\ 7.0;\ 8.0;\ 9.0;\ 10.0;\ 11.0;\ 12.0;\ 13.0;\ 14.0;\ 15.0\}.$$

The full survey of all possible variants would need checking of 12073600 combinations. The starting value for enumeration algorithm was fixed to 193 N.

The following heuristic rules have been applied. They precise simply that the cross-sectional areas of the bars to the left of the structure are to be greater than those to the right, according to the expected stress distribution.

rule 1: IF *downward_acting_loads*
THEN *flexion_of_the_structure.*

rule 2: IF *flexion_of_the_structure*
THEN *stress_increasing_to_the_left.*

rule 3: IF *stress_increasing_to_the_left*
THEN ($A_6 > A_5$ and $A_6 > A_4$ and $A_6 > A_3$ and $A_6 > A_2$ and $A_6 > A_1$)
and ($A_5 > A_4$ and $A_5 > A_3$ and $A_5 > A_2$ and $A_5 > A_1$)
and ($A_4 > A_2$ and $A_4 > A_1$)
and ($A_3 > A_2$ and $A_3 > A_1$).

The results of the optimization are given in Table 2. The number of equivalent solutions is given but only the first optimal variant is referenced.

Table 2. Optimal solutions for Case 2.

| constraints | stress | stress displ. | stress buckling | displ.stress buckling |
|---|---|---|---|---|
| Optimum weight [N] | 192.6144 | 294.5599 | 270.9489 | 309.4489 |
| cross-sectional areas | | * | | |
| $A_1$ [m$^2$] | 0.0001 | 0.0002 | 0.0004 | 0.0004 |
| $A_2$ [m$^2$] | 0.0002 | 0.0004 | 0.0004 | 0.0004 |
| $A_3$ [m$^2$] | 0.0004 | 0.0006 | 0.0005 | 0.0005 |
| $A_4$ [m$^2$] | 0.0005 | 0.0005 | 0.0005 | 0.0005 |
| $A_5$ [m$^2$] | 0.0006 | 0.0010 | 0.0007 | 0.0011 |
| $A_6$ [m$^2$] | 0.0008 | 0.0015 | 0.0008 | 0.0014 |
| Number of checked variants for standard enumeration | 5 | 118600 | 57825 | 175196 |
| Number of checked variants for knowledge-based enumeration | 3 | 13337 | 8411 | 17006 |

* 2 equivalent solutions exist

## 6. Conclusions and final remarks

It is seen from the presented examples that the knowledge-based approach implies an enormous reduction (with respect to the "standard" version) in the number of variants that must be checked to find the optimum. Even for a very simple knowledge base, the average economy with respect to the "standard enumeration" version was about 88%. It means that the constraints for only about 12% of the generated variants needed to be verified to reach the optimum. The way of dividing the structure into linking zones influences, of course, the optimal weight solution.

In the knowledge-based discrete optimization method, a domain-specific knowledge is an active component of the algorithm and is used to eliminate the *a priori* "incorrect" design variables sets. The inference mechanism contained in the knowledge module is able to extract and apply an information given in the explicit or implicit form, represented in a "natural language" of the problem. The separation of numeric and symbolic processing enables easy changes or modifications.

Including non-algorithmic and non-numerical ability into conventional optimum design programs can improve performances of the engineering-oriented optimization tools. The potential of symbolic computations applied to problems of discrete optimization has been emphasized, however an additional development work in this field is necessary. The further studies of the knowledge acquisition, formalization of heuristics or automated learning from the database of optimization examples are indispensable to code and utilise this knowledge properly. It is hoped that the knowledge-based approach used in conjunction with numerical techniques can considerably enhance other conventional optimization procedures.

## References


1. J. Bauer, *A survey of methods for discrete optimum structural design*, CAMES, 1, 27–38, 1994.
2. W. Gutkowski and J. Bauer [Eds.], *Discrete structural optimization*, IUTAM Symposium Zakopane, Poland, 31.08-03.09.1993, Springer Verlag 1993.
3. H. Eschenauer, J. Koski and A. Osyczka [Eds.], *Multicriteria design optimization, procedures and applications*, Springer Verlag, Berlin, Heidelberg, New York 1990.
4. M. Kleiber [Ed.], *Artificial intelligence in computational engineering*, Ellis Horwood Limited, 1990.
5. M. Balachandran, *Knowledge-based optimum design, topics in engineering*, C.A. Brebbia and J.J. Connor [Eds.], vol. 10, Computational Mechanics Publications, Southampton UK, Boston USA 1993.
6. Z. Iwanów, *An algorithm for finding an ordered sequence of a discrete linear function*, Control and Cybernetics, **19**, 3-4, 129–154, 1990.
7. M. Pyrz, *Symbolic computations approach in controlled enumeration methods applied to discrete optimization*, [in:] Discrete Structural Optimization, W. Gutkowski and J. Bauer [Eds.], IUTAM Symposium Zakopane, Poland, 31.08-03.09.1993, 221–227, Springer Verlag, 1993.


LABORATOIRE DE MÉCANIQUE DE LILLE,
ECOLE UNIVERSITAIRE d'INGÉNIEURS DE LILLE, FRANCE.

*Received November 25, 1994.*