Decision Making Support System for Multicriteria Discrete Optimization of Technical Products

Ivo Malakov\textsuperscript{1,a}, Velizar Zaharinov\textsuperscript{2,b,*} and Georgi Dinev\textsuperscript{3,c}

\textsuperscript{1,2,3}TU-Sofia, blvd. Kliment Ohridski 8, Sofia 1000, Bulgaria

\textsuperscript{a}ikm@tu-sofia.bg, \textsuperscript{b}vzaharinov@tu-sofia.bg, \textsuperscript{c}gdinev@tu-sofia.bg

Keywords: engineering design, decision making, decision making support system, multicriteria optimization, discrete optimization, technical products

Abstract. The paper presents a decision making support system for choosing an optimal variant of technical products. The problem is formulated and its characteristic features outlined. The decision making support system implements various approved methods for solving the formulated problem. Its capabilities are extended with an advanced algorithm, based on ant colony optimization, and integrated as a program module in the decision making support system. An example problem is solved, showing some of the capabilities of the system, and the integrated algorithm.

Introduction

Today’s technical products are multicomponent technical systems, characterized by a great number of parameters [1]. Their effectiveness is, in a significant way, determined during the design process.

Optimization problems are solved repeatedly throughout the whole design process of a technical system - optimization of principle, optimization of layout, and optimization of production [2]. These optimization problems are, in general, multicriteria optimization problems. Moreover, they are inherently discrete.

There are limited number of decision making support systems (DMSS) that can be used successfully during the design process of technical products. Examples of such DMSS are 1000Minds [3, 4], D-Sight [5], IND-NIMBUS [6], and other. These systems are used in various applications and offer many methods and tools for solving multicriteria decision making problems. Nevertheless, they lack tools that can fully address some of the specific features of the problem for choosing an optimal variant of a technical system, particularly: solving discrete multicriteria optimization problems with many possible variants (more than $10^8$), solving discrete multicriteria optimization problems with constraints regarding compatibility between the components that build the product, and in cases with polyfunctional building components.

The known DMSS can be used for choosing an optimal variant of a technical product, but most of them require the generation (building) of all designed variants for the product. These represent the compatible combinations of the product’s components. The requirement for generating the designed variants leads to a significant expenditure of time for problems related to technical products built from many components.

The aim of the current paper is to present a DMSS, supporting the decision making process when solving multicriteria discrete optimization problems, and the integration of an advanced algorithm, based on ant colony optimization, in this system.

Problem formulation

The choosing of an optimal variant of a technical product is related to solving the following problem:

For a given set of alternative variants of a technical product, i.e. defined alternative elementary devices executing its partial functions and the connections between the devices, define such a
compatible combination of the said devices (alternatives), which would be able to execute the
general function of the product, and would satisfy preliminary defined requirements and conditions
(constraints) related to the technical, economical, and other properties of the product.

Some of the characteristic features of the formulated problem are [7]: it is a vector (multicriteria,
multiobjective) optimization problem; great number \((10^9 - 10^{12})\) of possible variants; it belongs to
the class of discrete programming problems; constraints regarding compatibility (possibility for
combining and simultaneous work) of the elementary devices; different importance (weight,
significance) of the various evaluation criteria; possibility for defining constraints, high level of
uncertainty.

The general mathematical model for this problem has the following form [7]: find a variant
\(x^* = \{x_1^*, x_2^*, \ldots; x_n^*, \ldots; x_N^*\}\), \(l \leq L_n, n = 1 \div N\) optimal according to a set of criteria:

\[
\text{optF}(x) = \{f_k(x); k \in K\},
\]

with a predefined importance, and satisfying the constraints:

\[
g_m(x) \leq b_m, m \in M_1, \quad g_m(x) \geq b_m, m \in M_2,
\]

\[
x = \{x'_n; l \in L_n, n = 1 \div N\}, \quad x \in X, \quad x'_n \in X_n = \{x'_n, x^2_n, \ldots, x^n_n\}, |L_n| = l_n,
\]

where \(F(x)\) is the vector of objective functions; \(f_k(x)\) - \(k^{\text{th}}\) technical or economic characteristic of the
technical product, for which is sought optimal (maximum or minimum) value, \(k \in K\); \(g_m(x)\) - \(m^{\text{th}}\)
technical or economic characteristic of the technical product, over which value there is a constraint
of the kind \(\leq\) or \(\geq\); \(X\) - set of the possible variants; \(X_n\) - set of alternative elementary devices
executing the \(n^{\text{th}}\) partial function, \(n = 1 \div N\); \(x^l_n\) - \(l^{\text{th}}\) elementary device executing the \(n^{\text{th}}\) partial
function; \(N\) - the partial functions’ count for the given system; \(f_k(x), g_m(x)\) - functions of a discrete
argument, which are defined through tables.

The set \(X\) of variants for the construction of the technical product can be presented as a net model
(a directed graph) [7]. Every node (circle) of the model represents a particular elementary device
every arrow – a connection between two devices. In Fig. 1 is shown an example of a set of possible
variants of a technical product as a net model, where \(TF_N\) is the \(N^{\text{th}}\) partial function of the
technical product; \(L_N\) – the number of the alternative elementary devices, which execute the \(N^{\text{th}}\)
partial function. The possible combinations of the elementary devices into structures, which can
execute the general function of the technical product, are shown with arrows. Every path linking the
beginning (H) with the end (K) of the net model represents a possible variant.

**Algorithmic support**

For solving the problem of choosing an optimal variant of a technical product an advanced
algorithm for multicriteria optimization is developed. It is based on ant colony optimization (ACO)
[8, 9, 10]. As optimization model ACO is a multi-agent system in which the low level interaction
between the different agents (i.e. artificial ants) becomes a complex behavior of the whole ant
colony. The proposed algorithm applies the general structure of ant colony optimization for the
solution of the specific problem for choosing an optimal variant of a technical product. To that end a
concrete model (5) for preference calculation is developed and a procedure for pheromone
distribution is proposed.

The main logic of the algorithm is as follows:
- For every objective function an ant population is created;
- The ants from a particular population search for optimal solutions only for the objective
function appointed to their population;
The populations are independent of each other but the behavior of the ants is according to a unified model based on probability \( p_{nl}^k \) for choosing of elementary device \( x_{nl}^k \), \( k \in K \):

\[
p_{nl}^k = \frac{(\varphi_{nl}^t)^\alpha (d_{nl}^k)^\beta}{\sum_{l=1}^{L} (\varphi_{nl}^t)^\alpha (d_{nl}^k)^\beta},
\]

(4)

where \( \varphi_{nl}^t \) is pheromone level at node \( x_{nl}^t \) in moment of time \( t \); \( d_{nl}^k \) - the preference of elementary device \( x_{nl}^k \); \( \alpha \) and \( \beta \) - influence parameters, and

\[
d_{nl}^k = \left( I \pm \frac{\left( x_{nl}^t \right)^2_k}{\max \left( X_n^k \right)^2 + \min \left( X_n^k \right)^2} \right) \prod_{m=1}^{M} \left( I \mp \frac{\left( x_{nl}^m \right)^2}{\max \left( X_n^m \right)^2 + \min \left( X_n^m \right)^2} \right),
\]

(5)

where \( (x_{nl}^t)_k \) is the value of the elementary device’s \( x_{nl}^t \) parameter that corresponds to the objective function \( f_k(x) \); \( \max X_n^k \) and \( \min X_n^k \) - maximum and minimum value of the set \( X_n \) for objective function \( f_k(x) \); \( (x_{nl}^m)_m \) - the value of the elementary device’s \( x_{nl}^m \) parameter that corresponds to the constraint \( g_m(x) \), \( m \in M = M_1 \cup M_2 \); \( \max X_n^m \) and \( \min X_n^m \) – maximum and minimum value of the set \( X_n \) for constraint \( g_m(x) \).

- The objective functions are solved incrementally. This means that the ants from the different populations propose solutions by following a predefined ordering of the objective functions;
- An iteration finishes when all populations have generated solutions for their appointed objective function;
- After every iteration the found solutions are evaluated against an optimization criterion (min-max) using domination relations. These solutions are the starting point for the next iteration of the algorithm.

**Software support**

The software tool used for solving the multicriteria problem of choosing an optimal variant of a technical product is the DMSS PolyOptimizer [11]. A software implementation of the algorithm described in the previous section is developed and integrated in the modular structure of PolyOptimizer as a new module.

PolyOptimizer implements two methods for solving discrete optimization problems: full combinations method (FCM) and method for the consecutive analysis of variants [12] (MCAV), implemented respectively by the modules FCM Solver and MCAV Solver. The newly integrated, third method, is the algorithm based on ant colony optimization and is implemented by the ACO solver module. FCM is used for problems with smaller number of possible variants for the technical product. FCN can be used for solving of problems with up to \( 9^{10} \) variants. For a bigger number of variants the time needed for finding solution rises exponentially.
For solving problems with greater dimensionality than \(9^{10}\) combinations advanced algorithms of MCAV are used [7]. The method is characterized by a directed search for the optimal solution. It can reduce the number of possible variants and makes possible the application of FCM to larger problems.

For some complex problems it is not possible to find a solution even with MCAV because in some cases this method cannot converge. This is where the strength of the newly integrated algorithm lies. ACO is a metaheuristic method, and that means it does not guarantee finding the optimal solution, instead it guarantees that a close to the optimal solution would be found in a short period of time. Depending on the problem, ACO can even find the optimal solution.

PolyOptimizer has a module, GraphGUI, that is responsible for data input of problems with compatibility constraints between the elementary devices. It is a color coded graph through which the user inputs the relevant information. The modules Graph decomposition and LC Solver are concerned with solving of problems with compatibility constraints.

Example

The presented DMSS is used for choosing of an optimal variant of an automated assembly system. The assembled product is a portion of a handheld drill (Fig. 2), which portion is comprised of six different assemblies. A technological process is developed for assembling the assemblies. This process represents the general function of the designed technical system. The operations for assembling each assembly are grouped and are considered as one partial function of the system. The general function of the designed assembly system includes the following partial functions \(TF_n, n = 1 \div 6\):
- \(TF_1\) - assembling of assembly 1 comprised of parts 1, 5, 6, 7, 8, 9, and 10;
- \(TF_2\) - assembling of assembly 2 comprised of parts 3, 4, 11, 19, 37, and assembly 1;
- \(TF_3\) - assembling of assembly 3 comprised of parts 31, 32, 42, and 44;
- \(TF_4\) - assembling of assembly 4 comprised of parts 39, 40, and 41;
- \(TF_5\) - assembling of assembly 5 comprised of parts 12, 21 \(- 2\) pcs., and assembly 2;
- \(TF_6\) - assembling of assembly 6 comprised of parts 2, 16, 17, 18, 20, 22, 23, 24, 25 \(- 2\) pcs., 26, 27, 29, 38, and assemblies 3, 4, and 5.

Variants \(x_n, n = 1 \div 6, l \in L_n\), are developed for the execution of every partial function \(TF_n\). The set of possible variants \(X\) that execute the general function of the assembly system is presented in Fig. 3 as a net model. The polyfunctional devices (devices that execute more than one partial function of the system) \(x^f_i\) and \(x^s_i\) are represented as nodes, marked with two concentric circles, and are put in the third column. The latter corresponds to the partial function that is firstly executed by the polyfunctional devices. The number of possible variants of the designed assembly system is 3150.

The problem for choosing an optimal variant of the assembly system is solved. It has the following formulation: find a variant \(x^*_n\), executing the general function of the assembly system, for which:

\[
\min f_1(x) = \sum f_1(x^l_n), \quad \min f_2(x) = \sum f_2(x^s_n),
\]

satisfying the constraints:

\[
g_1(x) = \sum g_1(x^l_n) \leq 40, \quad g_2(x) = \sum g_2(x^s_n) \leq 22,
\]

where \(x \in X, \ x^l_n \in X_n, n = 1 \div 6, l \in L_n, |L_1|=5, |L_2|=5, |L_3|=5, |L_4|=4, |L_5|=3, |L_6|=3; \ X\) is the set of possible variants of the assembly system; \(X_n\) - the set of elementary devices \(x^l_n\), that execute the \(n^{th}\) partial function of the assembly system; \(f_1(x)\) - production cost, \$/1000 pcs.; \(f_2(x)\) - price of the automated assembly system, \$; \(g_1(x)\) - occupied area, m²; \(g_2(x)\) - number of operators.
The problem is solved with all three available algorithms – FCM, MCAV and ACO. Because the set of elementary devices includes polyfunctional elements, before applying the algorithms it is necessary to decompose the set of possible variants into subsets within which all elementary devices are compatible. For that purpose the PolyOptimizer’s modules GraphGUI and Graph decomposition are used. The set of possible variants is decomposed into two subsets. Every subset is solved with the three algorithms (Table 1). The solutions found with ACO coincide with the solutions found with the other two algorithms (for parameters $\alpha = \beta = 2$, $\varphi_n^\theta = 1$). According to min-max criterion the solution to the problem (6), (7) is solution number 2 (Table 1).

**Table 1. Solutions for the two subsets**

<table>
<thead>
<tr>
<th>№</th>
<th>Solution found with</th>
<th>$f_1(x)$</th>
<th>$f_2(x)$</th>
<th>Deviations</th>
<th>$g_1(x)$</th>
<th>$g_2(x)$</th>
<th>Variant</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FCM, MCAV, ACO</td>
<td>3841</td>
<td>793700</td>
<td>$w_1 \approx 0.047$</td>
<td>38.5</td>
<td>13.9</td>
<td>$x_1^3; x_2^3; x_3^4; x_4^5; x_5^2; x_6^2$</td>
</tr>
<tr>
<td>2</td>
<td>FCM, MCAV, ACO</td>
<td>3881</td>
<td>790000</td>
<td>$w_1 \approx 0.066$</td>
<td>38.5</td>
<td>14.1</td>
<td>$x_1^3; x_2^4; x_3^5; x_4^5; x_5^2$</td>
</tr>
</tbody>
</table>

**Conclusion**

The paper presents a decision making support system for choosing an optimal variant. The problem for choosing an optimal variant is repeatedly solved in the design process of technical products. The system has a modular build and implements advanced algorithms of appraborted methods for solving multicriteria combinatorial problems of the discrete programming - method of the consecutive analysis of variants and a method based on ant colony optimization. These methods
allow for solving of problems with big number of variants. Additionally, they are well suited for the specifics of the systematic design approach when designing technical products. This is a prerequisite for reducing the time needed for finding a solution, because the need for generating the possible variants of the whole product is eliminated. It is enough to develop the morphological table of the alternative variants executing the partial functions of the product, and to define the compatibility between these alternatives. The system also solves the problem related to presence of polyfunctional building elements and/or constraints regarding their compatibility when building technical products. In addition, the system has user-friendly graphical user interface, do not require knowledge in the used mathematical apparatus or in programming.

All this widens the capabilities of the proposed system compared to the known systems.

The efficiency of the proposed algorithm is tested by solving problems from the specialized literature, as well as with the full combinations method. The application of the system is illustrated with a concrete practical example for choosing of a variant of an automated assembly system.

The application field of the presented system is not limited to technical products. As it is based on the systematic approach for design of objects for which an optimal variant is sought, the system can be applied to a variety of other fields where decision making support is required.

Acknowledgments

The presented scientific research is financed from the Inner competition of TU-Sofia – 2016.

References

[12] V.S. Mihalevich, V.L. Volkovich, Computational methods for analysis and design of complex systems, Nauka, Moscow, 1982