Electronic scientific and practical journal

INTELLECTUALIZATION OF LOGISTICS AND SUPPLY CHAIN MANAGEMENT

#3 (2020)
October '20

WWW.SMART-SCM.ORG
ISSN 2708-3195
DOI.ORG/10.46783/SMART-SCM/2020-3

Electronic scientific and practical collection
INTELLECTUALIZATION OF LOGISTICS AND SUPPLY CHAIN MANAGEMENT

Electronic scientific and practical publication in economic sciences

ISSN 2708-3195
DOI: https://doi.org/10.46783/smart-scm/2020-3

Released 6 times a year

№ 3 (2020)
October 2020

Kyiv - 2020

Founder: Viold Limited Liability Company

The electronic scientific and practical journal is registered in international scientometric data bases, repositories and search engines. The main characteristic of the edition is the index of scientometric data bases, which reflects the importance and effectiveness of scientific publications using indicators such as quotation index, h-index and factor impact (the number of quotations within two years after publishing).

In 2020, the International Center for Periodicals (ISSN International Center, Paris) included the Electronic Scientific and Practical Edition "Intellectualization of Supply Chain Management" in the international register of periodicals and provided it with a numerical code of international identification: ISSN 2708-3195 (Online).

Recommended for dissemination on the Internet by the Academic Council of the Department of Logistics NAU (No. 7 of February 26, 2020). Released 6 times a year. Editions references are required. The view of the editorial board does not always coincide with that of the authors.

t.me/smart_scm
facebook.com/Smart.SCM.org
twitter.com/ScmSmart

DOI: https://doi.org/10.46783/smart-scm/2020-3
e-mail: support@smart-scm.org

тел.: (063) 593-30-41
https://smart-scm.org

Contents

EVENTS AND SCIENTIFIC CONFERENCES

UDC 004.023, 330.45
JEL Classification: C61, M 15.
Received: 13 October 2020

DOI: https://doi.org/10.46783/smart-scm/2020-3-1

Fedorov E. E. Doctor of Technical Science, Associate Professor, Professor of Department Robotics and Specialized Computer Systems, Cherkasy State Technological University (Ukraine)

ORCID – 0000-0003-3841-7373
Researcher ID – AAO-6744-2020
Scopus author id: 47161087200, 47161155900, 57205185819

Nikolyuk P. K., Doctor of Physics and Mathematics Sciences, Professor, Professor of Department Computer Sciences and Information Technologies, Vasil` Stus Donetsk National University (Ukraine)

ORCID – 0000-0002-0286-297X
Researcher ID – H-3223-2017
Scopus author id: –

Nechyporenko O. V., PhD, Associate Professor, Associate Professor of Department Robotics and Specialized Computer Systems, Cherkasy State Technological University (Ukraine)

ORCID – 0000-0002-3954-3796
Researcher ID –
Scopus author id: –

Chioma E. V., Student of Department Computer Sciences and Information Technologies, Vasil` Stus Donetsk National University (Ukraine)

ORCID – 0000-0002-0286-297X
Researcher ID –
Scopus author id: –

INTELLECTUALIZATION OF A METHOD FOR SOLVING A LOGISTICS PROBLEM TO OPTIMIZE COSTS WITHIN THE FRAMEWORK OF LEAN PRODUCTION TECHNOLOGY

Eugene Fedorov, Peter Nikolyuk, Olga Nechporenko, Esta Chioma. "Intellectualization of a method for solving a logistics problem to optimize costs within the framework of Lean Production technology". *In the article, within the framework of intellectualization of the Lean Production technology, it is proposed to optimize the costs arising from the insufficient efficiency of placing goods in the warehouse by creating an optimization method based on the immune metaheuristics of the T-cell model, which allows solving the knapsack constrained optimization problem. The proposed metaheuristic method does not require specifying the probability of mutation, the number of mutations, the number of selected new cells and allows using only binary potential solutions, which makes discrete optimization possible and reduces computational complexity by preventing permanent transformations of real potential solutions into intermediate binary ones* *and vice versa. An immune metaheuristic algorithm based on the T-cell model has been created, intended for implementation on the GPU using the CUDA parallel information processing technology. The proposed optimization method based on immune metaheuristics can be used to intellectualize the Lean Production technology. The prospects for further researches are to test the proposed methods on a wider set of test databases.*

***Keywords**:* lean manufacturing, immune metaheuristics, T-cell model, conditional optimization, knapsack problem.

Євген Федоров, Петро Ніколюк, Ольга Нечипоренко, Еста Чіома. "Інтелектуальна реалізація методу логістичного рішення для оптимізації витрат за технологією Lean Rroduction". *У статті в рамках інтелектуалізації технології Lean Rroduction пропонується оптимізувати витрати, що виникають в результаті відсутності ефективності розміщення товарів на складі, шляхом створення методу оптимізації на основі імунної метаевристики моделі Т-клітин, що дозволяє вирішити проблему умовної оптимізації про рюкзак. Запропонований метаеврістичний метод не вимагає задання ймовірності мутації, кількості мутацій, кількості відібраних нових клітин і дозволяє використовувати тільки бінарні потенційні рішення, що робить можливою дискретну оптимізацію і знижує обчислювальну складність шляхом запобігання постійній трансформації фізичних потенційних рішень в проміжні бінарні і зворотні. Створено імунний алгоритм метаевристики на основі моделі Т-клітин, призначений для впровадження на GPU за допомогою технології паралельної обробки інформації CUDA. Запропонований метод оптимізації на основі імунної метаевристики може бути використаний для інтелектуалізації технології Lean Rroduction. Перспективи подальших досліджень включають тестування запропонованих методів на більш широкому наборі тестових баз даних.*

Ключові слова: ощадливе виробництво, імунна метаевристика, модель Т-клітин, умовна оптимізація, задача про рюкзак.

Евгений Федоров, Петр Николюк, Ольга Нечипоренко, Эста Чиома. "Интеллектуализация метода решения логистической задачи для оптимизации затрат в рамках технологии Lean Production". *В статье в рамках интеллектуализации технологии Lean Production предлагается оптимизация затрат, возникающих вследствие недостаточной эффективности размещения товаров на складе, посредством создания метода оптимизации на основе иммунной метаэвристики модели Т-клеток, который позволяет решать задачу условной оптимизации о рюкзаке. Предложенный метаэвристический метод не требует задания вероятности мутации, количества мутаций, количества отбираемых новых клеток и позволяет использовать только бинарные потенциальные решения, что делает возможной дискретную оптимизацию и снижает вычислительную сложность за счет предотвращения постоянных преобразований вещественных потенциальных решений в промежуточные бинарные и обратно. Создан иммунный метаэвристический алгоритм на основе модели Т-клеток, предназначенный для реализации на GPU посредством технологии параллельной обработки информации CUDA. Предложенный метод оптимизации на основе иммунной метаэвристики может использоваться для интеллектуализации технологии Lean Production. Перспективы дальнейших исследований заключаются в тестировании предложенных методов на более широком наборе тестовых баз данных.*

Ключевые слова: бережливое производство, иммунная метаэвристика, модель Т-клеток, условная оптимизация, задача о рюкзаке.

Introduction. At present many worldwide companies are optimizing their business processes based on Lean Production technology. The concept of Lean Production is that it clearly identifies seven groups of costs that do not create value for final buyers, and therefore, the primary efforts of any company should be directed to minimizing these costs. However, the problem of finding models to minimize these costs is quite complicated and requires searching for the new solutions. As a result, the relevance of the development of methods for the intellectualization of Lean Production technology, which is based on the solution of optimization problems, significantly increases.

Literature and research review. Highly computationally complex optimization methods that find an accurate solution. Optimization methods that find an approximate solution through directional search have a high probability of hitting a local extremum. Random search methods do not guarantee convergence. Consequently, there is a problem of insufficient efficiency of optimization methods, which needs to be addressed.

Metaheuristics (or modern heuristics) [2-5] are used to find an accelerate quasi-optimal solution optimization problems and reduce the probability of hitting a local extremum. Metaheuristics empowered of heuristics by combining heuristic methods based on a high-level strategy [6-9].

The current metaheuristics have one or more of the following disadvantages:

- there is only an abstract description of the method or the description of the method is focused on solving only a certain problem [10];
- the influence of the iteration number on the process of finding a solution is not taken into account [11];
- the convergence of the method is not guaranteed [12];
- it is not possible to use non-binary potential solutions [13];
- the procedure for determining the values of parameters is not automated [14];
- it is not possible to solve the problems of conditional optimization [15];
- the lack of accuracy of the method [16].

In this regard, the problem of constructing effective metaheuristic optimization methods arises [17].

One of the popular metaheuristics are immune metaheuristics [18, 19], among which the T-cell model [20] can be distinguished, which allows solving constrained optimization problems.

Aims and Objectives. The aim of the work is to optimize the costs arising from the insufficient efficiency of placing goods in the warehouse by creating an optimization method based on immune metaheuristics that solves the knapsack problem.

To achieve the goal, the following tasks were put and decided:

1. Conduct an analysis of existing optimization methods aimed at optimizing costs within the framework of lean manufacturing technology.
2. Create an immune metaheuristic method based on the T-cell model for solving the knapsack problem.
3. Create an algorithm of the immune metaheuristic method based on the T-cell model, intended for implementation on the GPU using the CUDA technology.
4. Conduct a numerical study.

Results, analysis, and discussion. Optimization of costs associated with inefficient placement of goods in a warehouse can be reduced to the problem of a knapsack. To solve this problem, the work proposes an immune metaheuristic - a modified model of T cells that uses imitation of annealing.

As a function of the goal F, it is proposed to use the inverse function of income

$$F(x) = \left(\sum_{j=1}^{M} v_j x_j \right)^{-1} \to \min_x$$

where v_j is the income from the goods of the j -th type, defined,

w_j – weight of goods of the j -th type, defined,

x_j – goods presence of the j -th type (corresponds to the T-cell),

M – the number of types of goods.

As a limit, it is proposed to use the following function

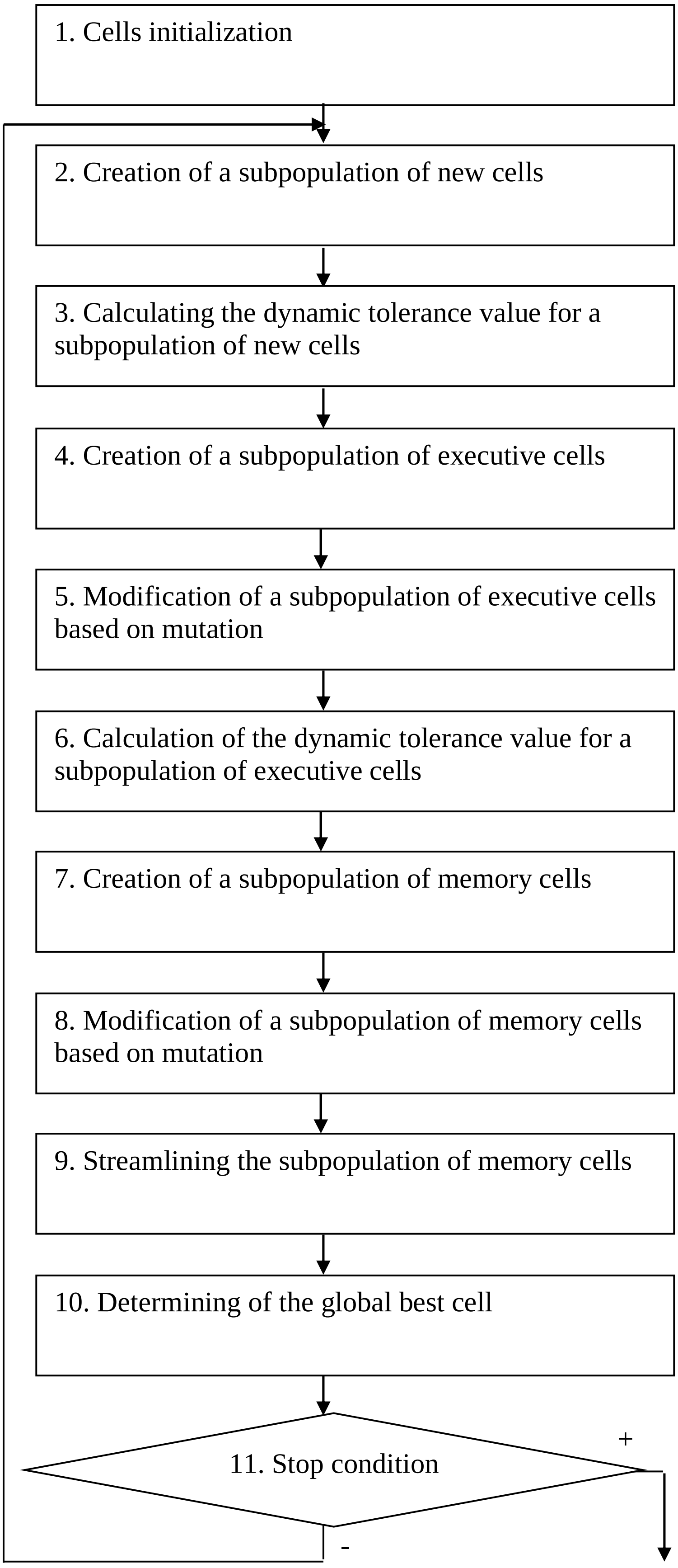

Figure 1 – The structure of the proposed immune metaheuristic method for solving the knapsack problem

$$g(x) = \max\left\{0, \sum_{j=1}^{M} w_j x_j - W\right\}$$

where W – is the maximum total weight of all goods, defined.

The structure of the proposed immune metaheuristic method is shown in Fig. 1.

The proposed metaheuristic method makes possible to find the quasi-optimal number of placed goods and consists of the following blocks:

Block 1 - Initialization:

– setting the number of the current iteration n to one;

– setting the number of iterations N ,;

– setting the cell length M ;

– setting the size of the subpopulation of new cells L_V ;

– setting the number of selected new cells, taking into account the restrictions $L1_V$ as $L1_V = L_V / 4$;

– setting the number of selected new cells without taking into account the restrictions $L2_V$ as $L2_V = L_V / 4$;

– setting the number of mutations of each executive cell N_E as $N_E = N$,

– setting the size of the subpopulation of memory cells L_M as $L_M = L_V / 4$;

– setting the number of mutations of each memory cell N_M as $N_M = N$,

– setting a static tolerance Δ_M for a subpopulation of memory cells;

– setting the probability of mutation of executive cells as $p^E = \dfrac{1}{M}$;

– setting the probability of mutation of memory cells as $p^M = \dfrac{1}{M}$;

– randomly create the best cell x^*

$$x^* = (x_1^*, ..., x_M^*),$$

$$x_j^* = \begin{cases} 1, & U(0,1) < 0.5 \\ 0, & U(0,1) \geq 0.5 \end{cases}$$

where $U(0,1)$ – is a function that returns a uniformly distributed random number in the range of $[0,1]$.

Block 2 – Creation of a subpopulation of new cells P^V

$$P^V = \{(x_k, s_k)\},\ k \in \overline{1, L_V}$$

$$x_k = (x_{k1}, ..., x_{kM}),$$

$$x_{kj} = \begin{cases} 1, & U(0,1) < 0.5 \\ 0, & U(0,1) \geq 0.5 \end{cases}$$

$$s_k = \max\{0, g(x_k)\}$$

Block 3 – Calculation of the dynamic tolerance value Δ_V for a subpopulation P^V

$$\Delta_V = \frac{1}{L_V} \sum_{k=1}^{L_V} s_k$$

If $\Delta_V < \Delta_M$, then $\Delta_V = 0.1$

Block 4 – Creation of a subpopulation of executive cells P^E with capacity L_E

4.1. Dividing a subpopulation of new cells P^V into a subset $P1^V = \{(x1_k, s1_k)\}$ containing cells for which $s1_k < \Delta_V$, and a subset $P2^V = \{(x2_k, s2_k)\}$ containing cells for which $s2_k \geq \Delta_V$.

4.2. Ordering the subset $P1^V$ by target function, i.e. $F(x1_k) < F(x1_{k+1})$

4.3. Ordering the set $P2^V$ by the sum of the values of all bounding functions, i.e. $s2_k < s2_{k+1}$

4.4. $L1_V$ of the first cells from an ordered set $P1^V$ and $L2_V$ the first cells from an ordered set $P2^V$ forms a subpopulation of executive cells $P^E = \{(x_i, s_i)\}$ with capacity $L_E = L1_V + L2_V$, while the first there are cells from the set $P1^V$

Block 5 – Modification of a subpopulation of executive cells P^E based on mutation

For each i - th cell is performed N_E, once as the following operations are performed:

– mutation

$r = U(0,1)$

$$\widehat{x}_{ij} = \begin{cases} 1, & (r < p^E \wedge x_{ij} = 0) \vee (r \geq p^E \wedge x_{ij} = 1) \\ 0, & (r < p^E \wedge x_{ij} = 1) \vee (r \geq p^E \wedge x_{ij} = 0) \end{cases}, \quad j \in \overline{1, M}$$

where $round()$ – is the function that rounds the number to the nearest integer.

– calculating the value of the constraint function

$$\widehat{s}_i = \max\{0, g(\widehat{x}_i)\}$$

– replacement by a mutant cell if the condition is met

If $\widehat{s}_i < s_i$ or $\widehat{s}_i = s_i \wedge F(\widehat{x}_i) < F(x_i)$, then $x_i = \widehat{x}_i$, $s_i = \widehat{s}_i$

Block 6 - Calculate the value of the dynamic tolerance Δ_E for a subpopulation P^E

$$\Delta_E = \frac{1}{L_E} \sum_{k=1}^{L_E} s_k$$

If $\Delta_E < \Delta_M$, then $\Delta_E = \Delta_M$

Block 7 - Creation of a subpopulation of memory cells P^M with capacity L_M

7.1. Dividing the subpopulation of executive cells P^E into a subset $P1^E = \{(x1_k, s1_k)\}$, containing cells for which $s1_k < \Delta_E$, and subset $P2^E = \{(x2_k, s2_k)\}$, containing cells for which $s2_k \geq \Delta_E$

7.2. Ordering the subset $P1^E$ by target function, i.e. $F(x1_k) < F(x1_{k+1})$

7.3. Ordering the set $P2^E$ by the sum of the values of all bounding functions, i.e. $s2_k < s2_{k+1}$

7.4. If $n = 1$, then L_M the first cells from the ordered union $P1^E \cup P2^E$ form a subpopulation of executive cells $P^M = \{(x_i, s_i)\}$

If $n > 1$, then $L_M / 2$ the first cells from the ordered union $P1^E \cup P2^E$ are replaced $L_M / 2$ by the worst (last) cells, a subpopulation of executive cells P^M

Block 8 - Modification of a subpopulation of memory cells P^M based on mutation

For each i - th cell is performed N_M, once as the following operations are performed:

– mutation

$r = U(0,1)$,

$$\widehat{x}_{ij} = \begin{cases} 1, & (r < p^M \wedge x_{ij} = 0) \vee (r \ge p^M \wedge x_{ij} = 1) \\ 0, & (r < p^M \wedge x_{ij} = 1) \vee (r \ge p^M \wedge x_{ij} = 0) \end{cases}, \; j \in \overline{1, M}$$

– calculating the value of the constraint function

$$\widehat{s}_i = \max\{0, g_z(\widehat{x}_i)\}$$

– replacement by a mutant cell if the condition is met

If $\widehat{s}_i < s_i$ or $\widehat{s}_i = s_i \wedge F(\widehat{x}_i) < F(x_i)$, then $x_i = \widehat{x}_i$, $s_i = \widehat{s}_i$

Block 9 - Ordering the subpopulation of memory cells P^M

Dividing the subpopulation of memory cells P^M into a subset $P1^M = \{(x1_k, s1_k)\}$ containing cells for which $s1_k < \Delta_M$, and a subset $P2^M = \{(x2_k, s2_k)\}$ containing cells for which $s2_k \ge \Delta_M$.

9.2. Ordering the subset $P1^M$ by target function, i.e. $F(x1_k) < F(x1_{k+1})$

9.3. Ordering the set $P2^M$ by the value of the bounding function, i.e. $s2_k < s2_{k+1}$

9.4. $P^M = P1^M \cup P2^M = \{(x_i, s_i)\}$

Block 10 – Determining of the global best cell

If $F(x_1) < F(x^*)$, then $x^* = x_1$

Block 11 - Stop Condition

If $n < N$, then increase the iteration number n by one and go to block 2.

For the proposed method, using the example of optimization of costs arising from insufficient efficiency of placing goods in a warehouse, an algorithm is considered intended for implementation on a GPU using the technology of parallel processing of information CUDA and shown in Fig. 2. This block diagram functions as follows.

Step 1 – Operator's input of the number of iterations N, the cell length M, the size of the subpopulation of new cells L_V, the number of selected new cells taking into account the limitations $L1_V$, the number of selected new cells without taking into account the limitations $L2_V$, the number of mutations of each executive cell N_E, the size of the subpopulation of memory cells L_M, the number of mutations of each memory cell N_M, static tolerance Δ_M for a subpopulation of memory cells, the probability of mutation of executive cells p^E, setting the probability of mutation of memory cells as p^M.

Step 2 – Randomly create the best cell x^*

Step 3 – The creation of a subpopulation of new cells P^V using GPU threads L_V that are grouped into 1 block. Each thread randomly creates a cell x_k and calculates the value of the bounding function for this cell s_k

Step 4 – Computation based on reduction of the dynamic tolerance Δ_V value for the subpopulation P^V across all cells using GPU threads L_V, which are grouped into 1 block. If $\Delta_V < \Delta_M$, then $\Delta_V = 0.1$

Step 5 – Dividing the subpopulation of new cells P^V into a subset $P1^V = \{(x1_k, s1_k)\}$ containing cells for which $s1_k < \Delta_V$, and a subset $P2^V = \{(x2_k, s2_k)\}$ containing cells for which $s2_k \ge \Delta_V$.

Step 6 – Ordering the subset $P1^V$ by target function, i.e. $F(x1_k) < F(x1_{k+1})$ using

GPU threads $|P1^V|$ which are grouped into 1 block

Step 7 – Ordering the subset $P2^V$ by the sum of the values of all bounding functions, i.e. $s2_k < s2_{k+1}$ using GPU threads $|P2^V|$ which are grouped into 1 block

Step 8 – $L1_V$ first cells from the ordered set $P1^V$ and $L2_V$ first cells from the ordered set $P2^V$ form a subpopulation of executive cells $P^E = \{(x_i, s_i)\}$ with capacity $L_E = L1_V + L2_V$, and the first cells from the set $P1^V$

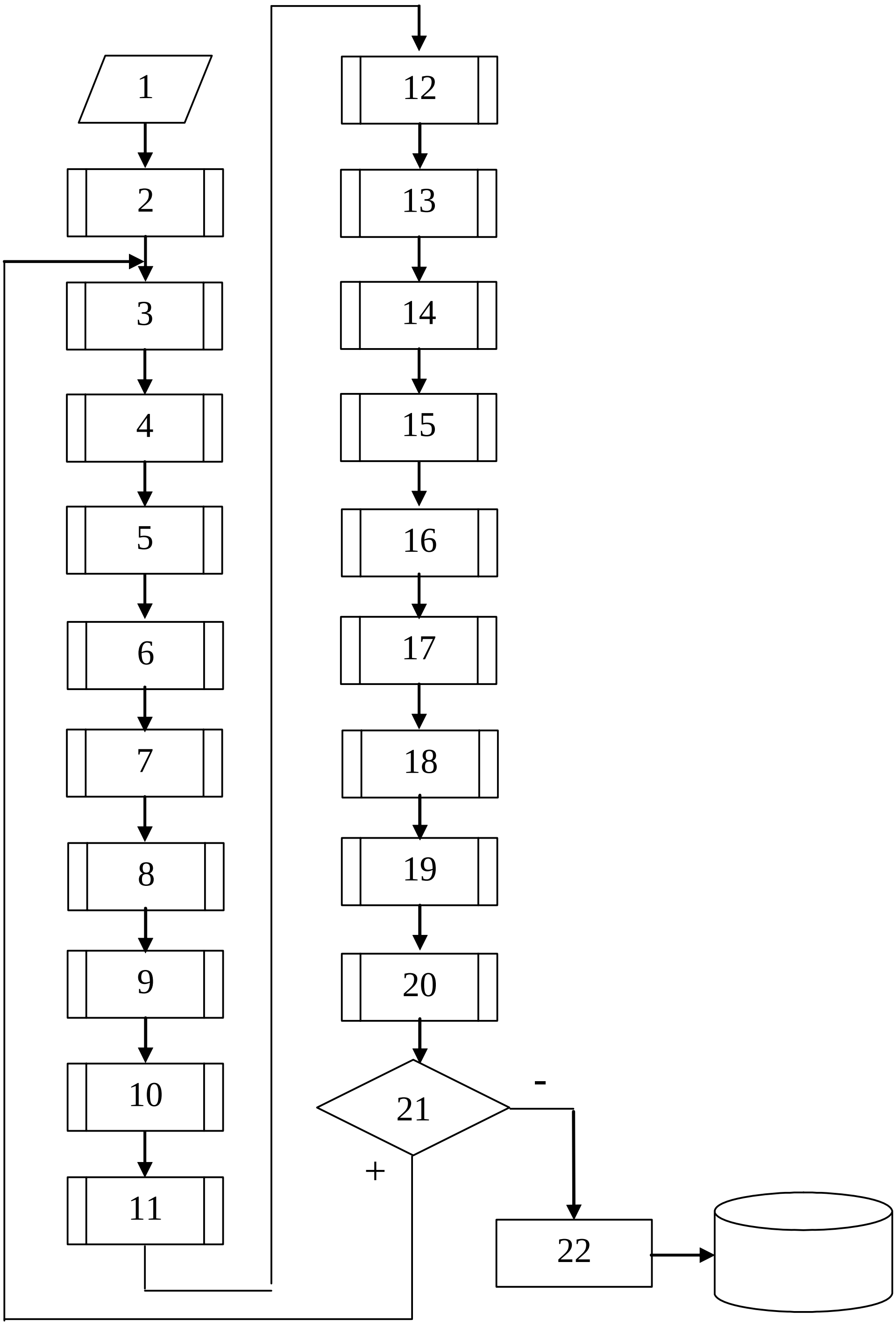

Figure 2 – Block diagram of the algorithm of the proposed immune metaheuristic method

Step 9 – Modification of a subpopulation of executive cells P^E based on mutation using GPU threads L_E that are grouped into 1 block. Each thread N_E once mutates a cell x_i and calculates the value of the limiting function for this cell s_i

Step 10 – Reduction computation of the dynamic tolerance value Δ_E for the subpopulation P^E across all cells using GPU threads L_E that are grouped into 1 block. If $\Delta_E < \Delta_M$, then $\Delta_E = \Delta_M$

Step 11 – Dividing the subpopulation of executive cells P^E into a subset $P1^E = \{(x1_k, s1_k)\}$ containing cells for which $s1_k < \Delta_E$, and a subset $P2^E = \{(x2_k, s2_k)\}$ containing cells for which $s2_k \geq \Delta_E$.

Step 12 – Ordering the subset $P1^E$ by target function i.e. $F(x1_k) < F(x1_{k+1})$ using GPU threads $| P1^E |$ which are grouped into 1 block

Step 13 - Ordering the set $P2^E$ by the sum of the values of all bounding functions, i.e. $s2_k < s2_{k+1}$ using GPU threads $| P2^E |$ which are grouped into 1 block

Step 14 – If $n = 1$, then L_M the first cells from the ordered union $P1^E \cup P2^E$ form a subpopulation of executive cells $P^M = \{(x_i, s_i)\}$, otherwise $L_M / 2$ first cells from the ordered union $P1^E \cup P2^E$ are replaced $L_M / 2$ the worst (last) cells, a subpopulation of executive cells P^M

Step 15 – Modification of a subpopulation of memory cells P^M based on the mutation using GPU threads L_M, which are grouped into 1 block. Each thread N_M once mutates a cell x_i and calculates the value of the limiting function for this cell s_i

Step 16 – Dividing the subpopulation of memory cells P^M into a subset $P1^M = \{(x1_k, s1_k)\}$ containing cells for which $s1_k < \Delta_M$, and a subset $P2^M = \{(x2_k, s2_k)\}$ containing cells for which $s2_k \geq \Delta_M$.

Step 17 – Ordering the subset $P1^M$ by target function i.e. $F(x1_k) < F(x1_{k+1})$ using GPU threads $| P1^M |$ which are grouped into 1 block

Step 18 – Ordering the set $P2^M$ by the value of the bounding function, i.e. $s2_k < s2_{k+1}$ using GPU threads $| P2^M |$ which are grouped into 1 block

Step 19 – Ordered sets $P1^V$ and $P2^V$ form a new subpopulation of memory cells P^M, i.e. $P^M = P1^M \cup P2^M = \{(x_i, s_i)\}$

Step 20 – Determining the global best cell according to the following rule

If $F(x_1) < F(x^*)$, then $x^* = x_1$

Step 21 – Stop Condition

If $n < N$, then increase the iteration number by one and go to step 4.

Step 22 – Writing the obtained global best position to the database.

In the work, the number of iterations N = 100, the size of the subpopulation of new cells L_V = 100, the number of selected new cells taking into account the constraints $L1_V$ = L_V / 4 = 25, the number of selected new cells without taking into account the constraints $L2_V$ == L_V / 4 = 25, the number of mutations of each executive cell N_E = N = 100, the size of the memory cell subpopulation $L_M == L_V / 4 = 25$, the number of mutations of each memory cell $N_M = N = 100$, the static tolerance $\Delta_M = 0.0001$ for the memory cell subpopulation.

For the knapsack problem, the search for a solution was carried out on the standard KNAPSACK_01 databases. For the proposed method, a root-mean-square error of 0.02 was obtained.

The traditional method for optimizing a T-cell model requires:

– setting the probability of mutation, the number of mutations, the number of selected new cells;

– real potential solutions, which makes discrete optimization impossible;

– constant transformations of real potential solutions into intermediate binary ones and vice versa.

The proposed method eliminates these disadvantages.

Conclusions.

1. To minimize losses that do not create consumer value and are the basis of Lean Production technology, an immune metaheuristic method based on the T-cell model was developed to solve the knapsack problem. The use of this method is aimed at minimizing costs arising from insufficient efficiency of the placement of goods in the warehouse.

2. The proposed metaheuristic method does not require setting the probability of mutation, the number of mutations, the number of selected new cells and allows using only binary potential solutions, which makes discrete optimization possible and reduces computational complexity by preventing constant transformations of real potential solutions into intermediate binary ones and back.

3. There was created an immune metaheuristic algorithm based on the T-cell model, intended for implementation on a GPU using the CUDA parallel processing technology.

4. The proposed optimization method based on immune metaheuristics can be used to intellectualize the Lean Production technology. Prospects for further research are in testing the proposed methods on a wider set of test databases.

References

1. Cluster Policy of Innovative Development of the National Economy: Integration and Infrastructure Aspects : monograph / under the editorship of professor Svitlana Smerichevska. Poznań: Wydawnictwo naukowe WSPIA, 2020. 380 p.

2. Talbi El-G. Metaheuristics: from design to implementation / El-G. Talbi. – Hoboken, New Jersey: Wiley & Sons, 2009. – 618 p.

3. Engelbrecht A.P. Computational Intelligence: an introduction / A.P. Engelbrecht. – Chichester, West Sussex: Wiley & Sons, 2007. – 630 p.

4. Yang X.-S. Nature-inspired Algorithms and Applied Optimization / X.-S. Yang. – Charm: Springer, 2018. – 330 pp.

5. Blum C. Hybrid Metaheuristics. Powerful Tools for Optimization / C. Blum, G. R. Raidl. – Charm: Springer, 2016. – 157 p.

6. Glover F. Handbook of Metaheuristics / F. Glover, G.A. Kochenberger. – Dordrecht: Kluwer Academic Publishers, 2003. – 570 p.

7. Yang X.-S. Optimization Techniques and Applications with Examples / X.-S. Yang. – Hoboken, New Jersey: Wiley & Sons, 2018. – 364 p.

8. Martí R., Handbook of Heuristics / R. Martí, P.M. Pardalos, M.G.C. Resende. – Charm: Springer, 2018. – 1289 p.

9. Gendreau M. Handbook of Metaheuristics / M. Gendreau, J.-Y. Potvin. – New York: Springer, 2010. – 640 p.

10. Doerner K.F. Metaheuristics. Progress in Complex Systems Optimization / K.F. Doerner, M. Gendreau, P. Greistorfer, W. Gutjahr, R.F. Hartl, M. Reimann. – New York: Springer, 2007. – 408 p.

11. Bozorg-Haddad O. Meta-heuristic and Evolutionary Algorithms for Engineering Optimization / O. Bozorg-Haddad, M. Solgi, H. Loaiciga. – Hoboken, New Jersey: Wiley & Sons, 2017. – 293 p.

12. Chopard B. An Introduction to Metaheuristics for Optimization / B. Chopard, M. Tomassini. – New York: Springer, 2018. – 230 p.

13. Radosavljević J. Metaheuristic Optimization in Power Engineering / J. Radosavljević. – New York: The Institution of Engineering and Technology, 2018. – 536 p.

14. Grygor O. Optimization method based on the synthesis of clonal selection and annealing simulation algorithms / Grygor O., Fedorov E., Utkina T., Lukashenko A., Rudakov K., Harder D., Lukashenko V. // Radio Electronics, Computer Science, Control. – 2019. – № 2. – P. 90-99

15. Fedorov E. Method for parametric identification of Gaussian mixture model based on clonal selection algorithm / Fedorov E., Lukashenko V.., Utkina T., Lukashenko A., Rudakov K. // CEUR Workshop Proceedings – 2019. – Vol. 2353. – P. 41-55

16. Alba E., Nakib A., Siarry P. Metaheuristics for Dynamic Optimization. – Berlin: Springer-Verlag, 2013. – 398 p.

17. Du K.-L. Search and Optimization by Metaheuristics. Techniques and Algorithms Inspired by Nature / K.-L. Du, M.N.S Swamy. – Charm: Springer, 2016. – 434 p.

18. Nakib A. Metaheuristics for Medicine and Biology / Nakib A., Talbi El-G. – Berlin: Springer-Verlag, 2017. – 211 p.

19. Subbotin S. Diagnostic Rule Mining Based on Artificial Immune System for a Case of Uneven Distribution of Classes in Sample / S. Subbotin, A. Oliinyk, V. Levashenko, E. Zaitseva // Communications. – Vol.3. – 2016. – P.3-11.

20. Brownlee J. Clever Algorithms: Nature-Inspired Programming Recipes / J. Brownlee. – Melbourne: Brownlee, 2011. – 436 p.