Routing Problems in VLSI Design: New Mathematical Models with Practical and Software Implementation

Oksana Pichugina[1], Olha Matsyi[2]

[1] *National Aerospace University "Kharkiv Aviation Institute", 17 Chkalova Street, Kharkiv, 61070, Ukraine*
[2] *V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine*

Abstract

This work is dedicated to the mathematical modelling of routing problems in VLSI design in the form of combinatorial, Euclidean combinatorial and nonlinear continuous optimization problems. The offered Euclidean models are tested on randomly generated samples by nonlinear IPOPT and APOPT optimization solvers implemented in Python.

Keywords

VLSI design, routing, computer chip, module, Euclidean combinatorial optimization, continuous polynomial optimization, ternary set

1. Introduction

The process of manufacturing a computer chip includes several stages, each of which is associated with some constraints such as economic, realizability, performance, power, signal integrity, reliability, and yield ones [1], [2], [3], [4]–[6], [7]. Solving placement, routing, timing, and many other problems are associated with the stages [6]. For instance, placement and routing problems, where placing modules on a chip is implemented, and a task of routing by connecting these modules into certain networks are considered, arise in economic, realizability, performance, reliability, and yield steps of VLSI design [6], [8], [9], [10].

This paper attacks the actual problem of routing in chip design. Normally, problems of this class are formulated as combinatorial optimization ones aiming to minimize the total length of routes between connected modules. In this case, combinatoriality arises from the natural constraints that the route cannot be arbitrarily laid on the chip, passing along some discrete grid plotted on it [2], [3], [6], [11-13].

In turn, combinatorial optimization problems exist in two fundamentally different formulations - combinatorial and Euclidean combinatorial [14-17]. In the first case (further combinatorial statements), the problem is formulated as an optimization problem over a combinatorial set, such as a set of permutations, partial permutations, or combinations [14], [17], [18]. In the late case (further Euclidean combinatorial statements), the problem is formulated as a discrete optimization problem, that is, an optimization program over a set of vectors in Euclidean space [15], [16].

Depending on the formulation chosen, the methods for solving routing problems differ significantly [4]. These are various approximate and heuristic methods utilizing combinatorial statements, including evolutionary and genetic algorithms [17], [20], [21], which provide a good solution in a reasonable time but does not guarantee its optimality [20].

When a Euclidean combinatorial statement is utilized, it becomes possible to accurately solve routing problems by discrete optimization methods, such as branch-and-bound and the cutting plane techniques [15], [18]. VLSI design is an expensive process that deserves appending time and other

COLINS-2022: 6th International Conference on Computational Linguistics and Intelligent Systems, May 12–13, 2022, Gliwice, Poland
EMAIL: oksanapichugina1@gmail.com (Oksana Pichugina); olga.matsiy@gmail.com (Olha Matsiy)
ORCID: 0000-0002-7099-8967 (Oksana Pichugina); 0000-0002-1350-9418 (Olha Matsiy)

machine resources to obtain an accurate solution. That is why the second group of methods is gaining more and more attention [18]. In this regard, there is also a need in developing new approaches to expanding the class of real-world problems for which Euclidean combinatorial statements are known. Among them is the theory of Euclidean combinatorial configurations [6], [15], [22-24]. There exist combinatorial and continuous Euclidean formulations of real problems. The latter formulate the optimization problem as a classical optimization problem with an objective function and analytic constraints. In this statement, methods of continuous nonlinear, sometimes even convex, programming can be applied to solve the problems under consideration [14], [25].

The theory of continuous functional representations (f-representations) of combinatorial sets is a new direction in Euclidean combinatorial optimization intended to construct analytic representations of combinatorial sets embedded in Euclidean space [16], [24]. This research area makes it possible to formulate combinatorial optimization problems, i.e., routing ones, as continuous programs, thus opening a possibility to solve them using previously inaccessible tools on nonlinear continuous optimization [14], [25]. This work is dedicated to the mathematical modelling of routing problems in the three indicated formulations – combinatorial, Euclidean combinatorial and Euclidean continuous, and verifying their validity and comparative analysis of the last two by nonlinear optimization tools implemented in Python.

2. Problem statement

We solve the following routing problem arising in chip design [4-6], [22]. There is a chip of length n and width m, where n, m are natural numbers. The chip contains modules forming a set of M including modules A and B. It is required to construct the shortest path between A and B, satisfying the following constraints:

- the path begins and ends at the metal wires' connection points of the modules A and B, while the metal wires' connection points of all the modules are placed at the nodes of the integer lattice:

$$P = J_m^0 \times J_n^0$$
$$where \quad J_m = \{1,...,m\}, J_m^0 = \{0\} \cup J_m;$$

- the path can pass only through nodes of the integer lattice;
- the path is formed from vertical and horizontal segments;
- the path does not cross the modules $M \setminus \{A,B\} = M'_{AB}$.

3. Mathematical modelling: combinatorial and Euclidean combinatorial formulations

Let us construct a mathematical model of the problem (further referred to as Model 1).

In order to define the path, it is sufficient to set its beginning

$$\left(x^s, y^s\right) \in P_A \subset P, \tag{1}$$

where P_A is a set of coordinates of the metal wires' connection points of A, as well as ternary vectors:

$$\begin{gathered} \bar{x}, \bar{y} : \bar{x} = \left(x_i\right)_{i \in J_k}; \bar{y} = \left(y_i\right)_{i \in J_k}; \\ x_i, y_i \in B^{\pm} = \{0, -1, 1\}, i \in J_k; \end{gathered} \tag{2}$$

k is the maximum route length between A and B.

In addition, for the route end, it should be:

$$\left(x^f, y^f\right) \in P_B \subset P, \tag{3}$$

where P_B is a set of the metal wires' connection points of B Points that form the path AB on the plane are

$$\left(x^s, y^s\right) = \left(x'_0, y'_0\right), \left(x'_1, y'_1\right), ..., \left(x'_k, y'_k\right) = \left(x^f, y^f\right),$$

where

$$\left(x'_1, y'_1\right) = \left(x'_0 + x_1, y'_0 + y_1\right) = \left(x^s + x_1, y^s + y_1\right);$$
$$\left(x'_2, y'_2\right) = \left(x^s + x_1 + x_2, y^s + y_1 + y_2\right);$$
$$\dots$$
$$\left(x^f, y^f\right) = \left(x^s + x_1 + ... + x_k, y^s + y_1 + ... + y_k\right).$$

On the whole,

$$\left(x'_i, y'_i\right) = \left(x^s + \sum_{j=1}^{i} x_j, y^s + \sum_{j=1}^{i} y_j,\right), i \in J_k^0. \tag{4}$$

Condition 3) is represented as follows:

$$x_i y_i = 0, \quad i \in J_k, \tag{5}$$

which in conjunction with (2) will guarantee movement toward B only in the vertical and horizontal directions.

Condition 4) now can be represented as

$$\left(x'_i, y'_i\right) \in P_{M'_{AB}}, \quad i \in J_k^0, \tag{6}$$

where $\left(x'_i, y'_i\right)$ satisfy the condition (4), where $P_{M'_{AB}}$ is the set of coordinates of points through which the path can be constructed. As a result, we obtain a combinatorial formulation of the problem in the form: find a set of points

$$\left(x^s, y^s\right), \left(x_i, y_i\right), i \in J_k, \tag{7}$$

such that

$$z = \sum_{i=1}^{k} \left(x_i^2 + y_i^2\right) \to min \tag{8}$$

subject to constraints (1)-(3), (5), (6), (8),

$$\left(x^f, y^f\right) = \left(x'_k, y'_k\right), \tag{9}$$

where $\left(x'_k, y'_k\right)$ is determined by the formula (6).

The objective function (8) expresses the route length, and our goal is to minimize it.

Note that taking into account (2), the condition (5) can be replaced by linear inequality:

$$x_i + y_i \le 1, \quad i \in J_k. \tag{10}$$

As a result, a new mathematical model (further referred to as Model 1') of the following form is formed: find the set (7) that is a solution to the optimization problem (8) under the constraints (1)-(3), (6)-(9).

From this combinatorial formulation, let us move to the construction of its Euclidean combinatorial formulation allowing its solution by means of partial integer nonlinear programming [25], [26-28].

We start with an analytic formalization of the condition (1) and represent it as follows:

$$\prod_{(x,y)\in A}\left(\left(x^s - x\right)^2 + \left(y^s - y\right)^2\right) = 0. \tag{11}$$

The expression in parentheses is 0 if and only if $\left(x^s, y^s\right) = \left(x, y\right) \in A$. Respectively, if (11) is satisfied, then (1) holds and vice versa.

Condition (3) can be represented analytically in a similar way:

$$\prod_{(x,y)\in B}\left(\left(x^f - x\right)^2 + \left(y^f - y\right)^2\right) = 0. \tag{12}$$

Finally, the expression (6) is rewritten as follows:

$$\begin{gathered}\prod_{(x,y)\in M'_{AB}}\left(\left(x'_i - x\right)^2 + \left(y'_i - y\right)^2\right) = 0, \\ 0 \le x'_i \le n, 0 \le y'_i \le n, i \in J_k^0.\end{gathered} \tag{13}$$

As a result, we get a mathematical model of our problem of the form (2), (4)-(9), (11)-(13) (further referred to as Model 2). Similarly to Model 1', Model 2' is formed from Model 2 by replacing condition (5) with the inequality (10), that is the problem (2), (4),(6)-(13).

Models 2, 2.1 are, in fact, polynomial partial integer programming problems, to which the corresponding nonlinear optimization methods are applicable, [14], [24], [25], [28], [29].

Finally, these models can be easily transferred into a continuous nonlinear optimization problem by replacing condition (2) with the following expressions:

$$x_i\left(x_i - 1\right)\left(x_i + 1\right) = 0, y_i\left(y_i - 1\right)\left(y_i + 1\right) = 0, i \in J_k,$$

which simplification results in:

$$x_i^3 - x_i = 0, y_i^3 - y_i = 0, i \in J_k. \tag{14}$$

Model 3 – this will be a problem (4)-(9), (11)-(14), Model 3' – this will be a problem (4), (6)-(14).

Let us generalize the designed mathematical models to the case if the modules A and B as well as other pairs of modules need pairwise interconnection, while our task is to find the shortest total path connecting the modules. In this case, in addition to the above Constraints 1-4, the condition of non-intersection of the connection paths (further referred to as Condition 5) need to be added.

We add an index $l \in J_L$ to the model to reflect the order number of a pair of connected modules. As a result, we have a generalization of Model 1 (further referred to as Model 1.1).

Let

- $\left\{A_l, B_l\right\}_{l\in J_L}$ be pairs of connected modules, $M'_{A_l B_l} = M \setminus \left\{A_l, B_l\right\}, l \in J_L$,

- P_{A_l}, P_{B_l} be a set of coordinates of the legal module metal wires' connection points A_l, B_l, respectively ($l \in J_L$).
- $P_{M'_{A_l},B_l}$ is a set of free nodes of the lattice P for paving a path between A_l, B_l ($l \in J_L$).

Model 1.1 has the following form: find

$$\left(x^{sl}, y^{sl}\right), \overline{x}^l, \overline{y}^l, \left(x^{fl}, y^{fl}\right),\ l \in J_L, \tag{15}$$

such that

$$\overline{x}^l = \left(x_i^l\right)_{i \in J_k},\ \overline{y}^l = \left(y_i^l\right)_{i \in J_k},$$

being a solution to the optimization problem

$$z = \sum_{l=1}^{L}\sum_{i=1}^{k}\left(\left(x_i^l\right)^2 + \left(y_i^l\right)^2\right) \to min \tag{16}$$

subject to constraints:

$$\left(x^{sl}, y^{sl}\right) \in P_{A_l}, l \in J_L; \tag{17}$$

$$\left(x^{fl}, y^{fl}\right) \in P_{B_l}, l \in J_L; \tag{18}$$

$$x_i^l, y_i^l \in B^{\pm}, i \in J_k, l \in J_L; \tag{19}$$

$$x_i^l y_i^l = 0,\quad i \in J_k, l \in J_L; \tag{20}$$

$$\left(x^{'il}, y^{'il}\right) \in P_{M'_{A_l}B_l}, i \in J_k, l \in J_L; \tag{21}$$

$$D_l \bigcap D_{l'} = \varnothing, l \neq l', l, l' \in J_L, \tag{22}$$

where $\forall l \in J_L$

$$\left(x^{fl}, y^{fl}\right) = \left(x_k^{'il}, y_k^{'il}\right); \tag{23}$$

$$\left(x^{'il}, y^{'il}\right) = \left(x^{sl} + \sum_{j=1}^{i} x_j^l, y^{sl} + \sum_{j=1}^{i} y_j^l\right), i \in J_k; \tag{24}$$

$$D_l = \left\{ \left(x_i^{il}, y_i^{il} \right) \right\}_{i \in J_{k-1}}. \tag{25}$$

Here, D_l expresses the set of way-nodes between A_l and B_l, excluding its starting and ending points.

Let us proceed to the formation of a generalization of Model 2 to the case of connecting several modules. It is required to find a set (15), such that the minimum function (16) is attained and the constraints (19), (20) hold, $\forall\, l \in J_L$

$$\prod_{(x,y)\in A_l} \left(\left(x^{sl} - x \right)^2 + \left(y^{sl} - y \right)^2 \right) = 0; \tag{26}$$

$$\prod_{(x,y)\in B_l} \left(\left(x^{fl} - x \right)^2 + \left(y^{fl} - y \right)^2 \right) = 0; \tag{27}$$

$$\prod_{(x,y)\in M'_{A_l B_l}} \left(\left(x_i^{'l} - x \right)^2 + \left(y_i^{'l} - y \right)^2 \right) = 0, i \in J_k; \tag{28}$$

$$\prod_{i,i'=1}^{k} \left(\left(x_i^l - x_{i'}^{l'} \right)^2 + \left(y_i^l - y_{i'}^{l'} \right)^2 \right) \geq 1,\ l,l' \in J_L. \tag{29}$$

Here, the condition (29) is an analytic expression of the constraints (22), while (26)-(28) are a direct generalization of conditions (11)-(13).

Finally, replacing condition (19) with the following expression,

$$\left(x_i^l \right)^3 - x_i^l = 0, \left(y_i^l \right)^3 - y_i^l = 0,\ i \in J_k, l \in J_L, \tag{30}$$

we get Model 3.1 as a generalization of Model 3.

Like (5), (10), the linear analogue of (10) is the condition:

$$x_i^l + y_i^l \leq 1,\ \ i \in J_k,\ \ l \in J_L, \tag{31}$$

Taking into account (10), the condition (29) can be replaced by a linear constraint:

$$\sum_{i,i'=1}^{k} \left(\left(x_i^l - x_{i'}^{l'} \right)^2 + \left(x_{i'}^{l'} - y_{i'}^{l'} \right)^2 \right) \geq C_k^2,\ l,l' \in J_L, \tag{32}$$

as a result, we get another version of the model (further referred to as Model 3.1').

Remark 1 In the constructed models, the number k expresses the upper bound on the path length between two connected modules. For reducing the dimension of the problem, it is advisable to conduct a preliminary study to estimate the value of k for specific pairs of modules and replace k with the corresponding found values $k_l \leq k$, $l \in J_L$ for the pairs.

Remark 2 In the constructed models, it is easy to find the lower bound for the value of the objective function using the distance in the space L_∞. Thus, the objective function (8) can be estimated as follows:

$$
\begin{aligned}
& z \geq lb_{AB} = \|A - B\|_\infty = \\
= \; & \min_{\substack{(x,y)\in A \\ (x',y')\in B}} \max\{|x - x'| + |y - y'|\}.
\end{aligned} \tag{33}
$$

4. Software implementation and applications

Models 2.1, 3.1, 2.1', 3.1' were implemented in the Python software environment, using the nonlinear optimization package GEKKO [30], all models were tested using the APOPT nonlinear partial integer programming solver (Advanced Process OPTimizer) [31], while Models 2.1', 3.1', were also were solved using the IPOPT nonlinear partial integer programming solver (Interior Point OPTimizer) [23]. On average, IPOPT showed a certain advantage over the results of APOPT on randomly generated problems of dimension $m,n,L \in J_{10}$, where the area under the modules is 60 %-80 %.

Figure 1 shows an illustration for Models 1-3, where $L=5$, $m=5$, $n=7$, Modules A, B are highlighted in bold. Also, two paths I, II of length 6, 8 respectively, are depicted. Namely, when choosing, $k=8$, we have:

$$
\begin{aligned}
& I:(x^s, y^s) = (1,1), (x_i, y_i)_{i \in J_k} = ((0,1),(0,1),(1,0), \\
& (1,0),(0,1),(1,0),(0,0),(0,0)), \\
& z^I = 6.
\end{aligned}
$$

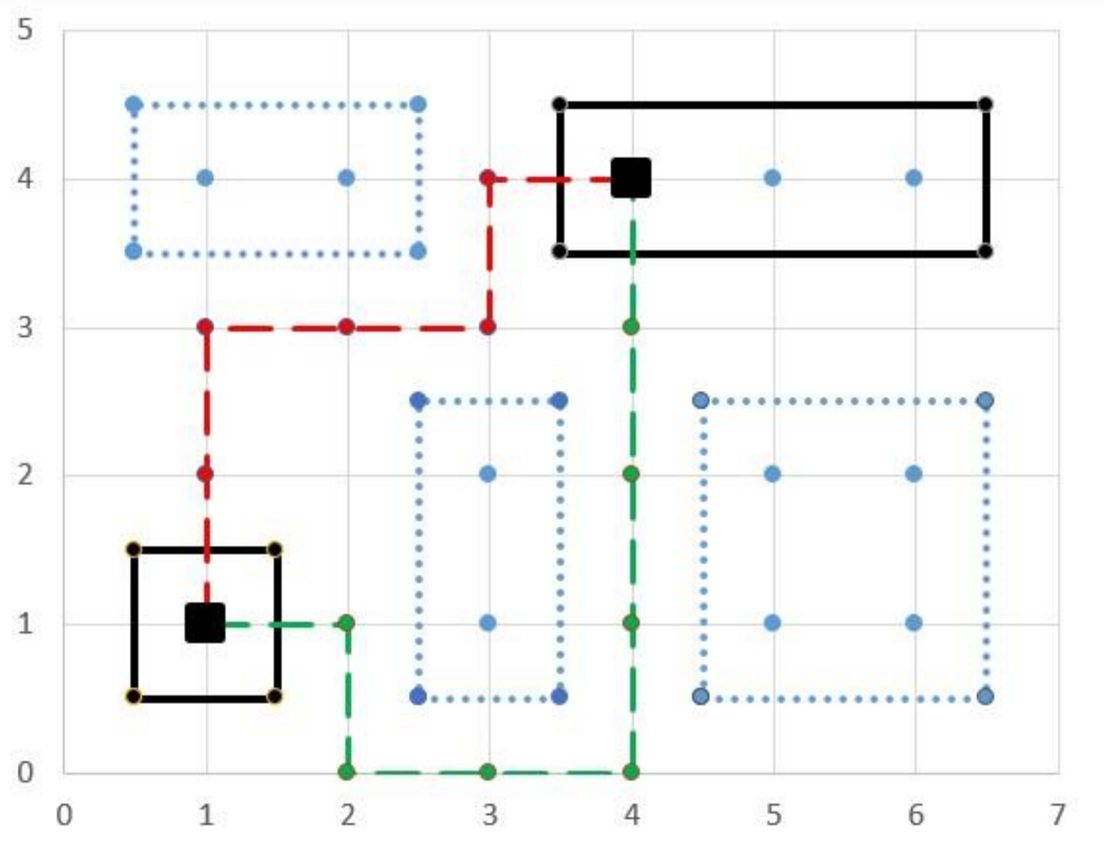

Figure 1: Example illustration

This solution corresponds to the path

$$\left(x_i^{'},y_i^{'}\right)_{i\in J_8}=(\left(1,1\right),\left(1,2\right),\left(1,3\right),\left(2,3\right),\left(3,3\right),\left(3,4\right),$$
$$\left(4,4\right),\left(4,4\right),\left(4,4\right)). \quad (34)$$

$$II:\left(x^s,y^s\right)=\left(1,1\right),\left(x_i,y_i\right)_{i\in J_k}=(\left(1,0\right),\left(0,-1\right),$$
$$\left(1,0\right),\left(1,0\right),\left(0,1\right),\left(0,1\right),\left(0,1\right),\left(0,1\right)),$$
$$z^{II}=8.$$

This solution corresponds to a path

$$\left(x_i^{'},y_i^{'}\right)_{i\in J_8}=(\left(1,1\right),\left(2,1\right),\left(2,0\right),\left(3,0\right),\left(4,0\right),\left(4,1\right),$$
$$\left(4,2\right),\left(4,3\right),\left(4,4\right)). \quad (35)$$

The (35) path is optimal because

$$lb_{AB}=\left\|A-B\right\|_\infty=\left\|\left(1,1\right)-(4,4)\right\|_\infty=6.$$

According to (33), the objective function reaches its lower bound, respectively, I is the optimal solution to the problem.

5. Conclusion

New routing models of practical problems of VLSI design are offered constructed by means of theories of Euclidean combinatorial configurations and f-representations of combinatorial sets mapped into Euclidean space. The models are validated by IPOPT and APOPT solvers, and an illustrative example is given.

6. References

[1] E. Bolotin, I. Cidon, R. Ginosar, and A. Kolodny, Cost considerations in network on chip, Integration, the VLSI Journal, Vol. 38, No. 1, Oct. 2004, Pp. 19–42. doi: 10.1016/j.vlsi.2004.03.006.

[2] J. Haase, Models, Methods, and Tools for Complex Chip Design: Selected Contributions from FDL 2012. Switzerland: Springer Science & Business Media, 2013.

[3] S. Held, D. Muller, D. Rotter, R. Scheifele, V. Traub, and J. Vygen, Global Routing with Timing Constraints, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2017, Pp. 1–14. doi: 10.1109/TCAD.2017.2697964.

[4] B. Korte, D. Rautenbach, and J. Vygen, BonnTools: Mathematical Innovation for Layout and Timing Closure of Systems on a Chip', Proceedings of the IEEE, Vol. 95, No. 3, Pp. 555–572, Mar. 2007. doi: 10.1109/JPROC.2006.889373.

[5] B. Korte, and J. Vygen, Combinatorial Problems in Chip Design, in Building Bridges, Springer, Berlin, Heidelberg, 2008, Pp. 333–368.

[6] L. Lavagno, I. L. Markov, G. Martin, and L. K. Scheffer, Eds., Electronic Design Automation for IC Implementation, Circuit Design, and Process Technology, 2 edition. Boca Raton: CRC Press, 2016.

[7] J. Williams and D. Thomas, Digital VLSI Design with Verilog: A Textbook from Silicon Valley Technical Institute, 2008 edition. New York: Springer, 2008.

[8] T. F. Chan, J. Cong, J. R. Shinnerl, K. Sze, M. Xie, and Y. Zhang, Multiscale Optimization in VLSI Physical Design Automation, in Multiscale Optimization Methods and Applications, W. W. Hager, S.-J. Huang, P. M. Pardalos, and O. A. Prokopyev, Eds. Springer US, 2006, Pp. 1-67.
[9] W. Nye, D. C. Riley, A. Sangiovanni-Vincentelli, and A. L. Tits, 'DELIGHT.SPICE: an optimization-based system for the design of integrated circuits', IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 7, No. 4, Apr. 1988, Pp. 501–519. doi: 10.1109/43.3185.
[10] V. P. Shmerko, Malyugin's Theorems: A New Concept in Logical Control, VLSI Design, and Data Structures for New Technologies, Automation and Remote Control, Vol. 65, No. 6, Jun. 2004, Pp. 893-912. doi: 10.1023/B:AURC.0000030902.91316.b9.
[11] K. Kawarabayashi and Y. Kobayashi, The Induced Disjoint Paths Problem, in Integer Programming and Combinatorial Optimization, A. Lodi, A. Panconesi, and G. Rinaldi, Eds. Springer Berlin Heidelberg, 2008, Pp. 47–61.
[12] O. Pichugina, Placement problems in chip design: Modeling and optimization, 2017 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T), 2017, Pp. 465–473. doi: 10.1109/infocommst.2017.8246440.
[13] O. Pichugina and O. Kartashov, Signed Permutation Polytope Packing in VLSI Design, in 2019 IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM), Feb. 2019, Pp. 4/50-4/55. doi: 10.1109/CADSM.2019.8779353.
[14] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, Nonlinear Programming: Theory and Algorithms, 3 edition. Hoboken, N.J: Wiley-Interscience, 2006.
[15] O. Pichugina, O. Matsyi, Boolean Satisfiability Problem: Discrete and Continuous Reformulations with Applications, in: 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), Oct. 2020, Pp. 623-627. doi:10.1109/TCSET49122.2020.235507.
[16] O. Pichugina and S. Yakovlev, Quadratic Optimization Models and Convex Extensions on Permutation Matrix Set, in Advances in Intelligent Systems and Computing IV, Nov. 2019, Pp. 231–246. doi: 10.1007/978-3-030-33695-0_17.
[17] S. Yakovlev, O. Kartashov, and O. Pichugina, Optimization on Combinatorial Configurations Using Genetic Algorithms, in Proceedings of the Second International Workshop on Computer Modeling and Intelligent Systems (CMIS-2019), Zaporizhzhia, Ukraine, Apr. 2019, Pp. 28–40. Available: http://ceur-ws.org/Vol-2353/paper3.pdf.
[18] P. Pardalos, D.-Z. Du, and R. L. Graham, Eds., Handbook of Combinatorial Optimization, 2nd ed. New York: Springer-Verlag, 2013.
[19] M. F. Anjos and F. Liers, Global approaches for facility layout and VLSI floorplanning, presented at the Handbook on semidefinite, conic and polynomial optimization, 2012.
[20] Teofilo F. Gonzalez, Ed., Handbook of Approximation Algorithms and Metaheuristics, Second Edition. Routledge Handbooks Online, 2018.
[21] S. Yakovlev, O. Kartashov, O. Pichugina, and K. Korobchynskyi, Genetic Algorithms for Solving Combinatorial Mass Balancing Problem, in 2019 IEEE 2nd Ukraine Conference on Electrical and Computer Engineering (UKRCON), Jul. 2019, Pp. 1061–1064. doi: 10.1109/UKRCON.2019.8879938.
[22] L. Koliechkina and O. Pichugina, A Horizontal Method of Localizing Values of a Linear Function in Permutation-Based Optimization, in Optimization of Complex Systems: Theory, Models, Algorithms and Applications, Cham, 2020, Pp. 355–364. doi: 10.1007/978-3-030-21803-4_36.
[23] A. Wächter and L. T. Biegler, On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming. 2004.

[24] Y. G. Stoyan, and S. V. Yakovlev, Theory and Methods of Euclidian Combinatorial Optimization: Current Status and Prospects, Cybern Syst Anal, Vol. 56, No. 3, May 2020, Pp. 366–379. doi: 10.1007/s10559-020-00253-6.
[25] J. Nocedal and S. Wright, Numerical Optimization, 2nd ed. New York: Springer-Verlag, 2006.
[26] T. Achterberg, T. Berthold, T. Koch, and K. Wolter, Constraint Integer Programming: A New Approach to Integrate CP and MIP, in Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, L. Perron and M. A. Trick, Eds. Springer Berlin Heidelberg, 2008, Pp. 6–20.
[27] F. A. Al-Khayyal, and P. M. Pardalos, Global optimization algorithms for VLSI compaction problems, Arabian J. Sci. Engrg., Vol. 16, No. 2, B, 1991, Pp. 335–339.
[28] R. K. Brayton, G. D. Hachtel, and A. L. Sangiovanni-Vincentelli, A survey of optimization techniques for integrated-circuit design, Proceedings of the IEEE, Vol. 69, No. 10, Oct. 1981, Pp. 1334–1362. doi: 10.1109/PROC.1981.12170.
[29] M. F. Anjos, and J. B. Lasserre, Eds., Handbook on semidefinite, conic and polynomial optimization, 2012th edition. Springer, New York, 2012.
[30] L. D. R. Beal, D. C. Hill, R. A. Martin, and J. D. Hedengren, GEKKO Optimization Suite, Processes, Vol. 6, No. 8, Aug. 2018, P. 106. doi: 10.3390/pr6080106.
[31] APMonitor, APMonitor/apopt, 2020. URL: https://github.com/APMonitor/apopt.