Genetic Algorithms in the Matrix Arrangement of Elements in Blocks on a Crystal Model

Vladislav Ivanovich Danilchenko
Computer-aided design of department, postgraduate

Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education "Southern Federal University".

Taganrog, Russia.
vdanilchenko@sfedu.ru

Yevgenia Vladimirovna Danilchenko
Computer-aided design of department, postgraduate

Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education "Southern Federal University".

Taganrog, Russia.
lipkina@sfedu.ru

Viktor Mikhailovich Kureichik
Computer-aided design of department, postgraduate

Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education "Southern Federal University".

Taganrog, Russia.
vmkureychik@sfedu.ru

***Abstract* — The article describes the solution to the problem of placing elements in blocks on a crystal model based on a genetic algorithm (GA), which allows creating an algorithmic environment in the field of genetic search for solving the problem of placing elements on a crystal, taking into account the criterion for the maximum number of linear segments. The purpose of this work is to find ways of placing elements in blocks based on a genetic algorithm. The scientific novelty lies in the development of a modified genetic algorithm for computer-aided design of very large-scale integrated circuits, taking into account the criterion for the maximum number of line segments. The problem statement in this work is as follows: to optimize the placement of elements in blocks by using a modified GA taking into account the criterion for the maximum number of line segments. A fundamental difference from the known approaches in the use of new modified genetic structures in computer-aided design, in addition, the paper presents a new method for placing elements on a crystal, based on a modified GA, taking into account the criterion for the maximum number of linear segments. Thus, the problem of creating methods, algorithms and software for automated placement of elements on a chip is currently of particular relevance.**

***Keywords* — genetic algorithms, graphs and hypergraphs, evolutionary computing, design automation, CAD, electronic production.**

I. Introduction

The main requirement when placing fragments on a chip is to create conditions that ensure complete routing of connections. It is necessary to take into account the characteristic properties of the crystal structure that have a significant impact on the results of the tracing. First of all, it should be noted that the number of vertical lines is approximately twice the number of horizontal ones. Therefore, it is preferable to implement connections with vertical segments. Moreover, from the point of view of the most economical use of the crystal resource, linear segments can be considered optimal. According to [1], the problem of placing elements in blocks on a crystal model belongs to NP complete problems [2], which means that in the general case, its solution, in principle, cannot be found in a finite time by any algorithm. To eliminate this problem, modified versions of the GA were developed, which take into account the criteria for the maximum number of linear segments.

Line segments represent the shortest implementation of connections, do not contain vials, do not occupy horizontal backbones. In addition, when using the most common two-layer routing technique, where all horizontal segments are in one layer and vertical segments are in another, and the route pivots are not located on the vertical trunks passing through the pins of other routes in this channel, line segments do not impose any restrictions. for the implementation of the remaining routes. The above considerations are also confirmed by practical experience in the design of very large scale integrated circuits of this type [3].

II. Problem Statement

Obtaining a matrix arrangement of elements in blocks on a crystal model, which is an orthogonal lattice, at the nodes of which it is required to place the vertices that represent the internal blocks. The number of rows of this matrix is equal to the number of Q lines on the crystal, and the number of columns is [N / Q], where N is the number of blocks. The criterion for the optimality of the transition from the force placement to the matrix one is the minimum change in the coordinates of the centers of the blocks and the maximum of the vertical edges. To solve this problem, a constructive genetic algorithm for obtaining an initial variant and iterative algorithms for improving placement are used. The initial placement algorithm is based on a sequential principle with the installation at the next step either one element or a group of blocks that form a "vertical chain". So, for the example under consideration "Fig. 1, b", at the first step, a group of blocks B5, B4, B1 is set in the first column, and at the second - B6, B2, B3 "Fig. 2, a" [3, 4].

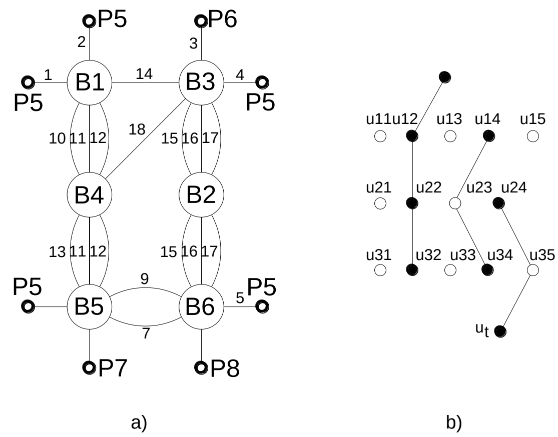

Fig. 1. – Initial variant of placement (a), and its model T (b)

The constraints of the problem are: requirements for the formation of a separate block from each peripheral element; fulfillment of inequality $\Delta_i \leq \Delta$ for each block i, Δ_i – total length of block elements. As a criterion for optimal placement, it is advisable to use the maximum number of internal chains of the block. This criterion can be considered justified, firstly, because it is associated with the minimum

978-1-6654-3941-1 @2021 IEEE

total length of the joints, and secondly, with this criterion, the description of the entire circuit with respect to the sides.

Therefore, in addition to the traditional criterion of optimal placement (minimum of the total length of the connection), when placing the elements on the chip, it is necessary to take into account the criterion of the maximum number of LS [4].

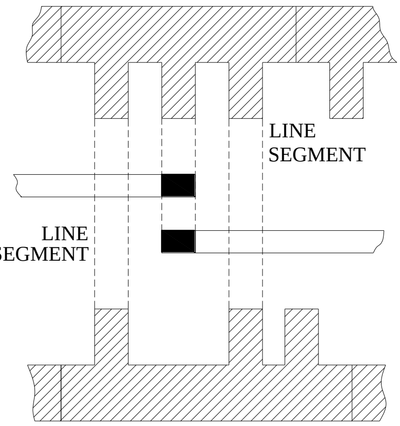

Fig. 2. – Examples of linear segments on the crystal model are given

Equation (1) reveals the necessary conditions for performing a complete trace that can be taken into account during the deployment phase. The first of them limits the number of elements and transit routes in each line [5].

$$\sum_{i=E} L_i + \langle (n_t - n_p) l_0 / p_0 \rangle \leq L, \qquad (1)$$

where E – many items placed in a ruler; L_i – element length i; n_t – number of transit routes, crossing the ruler; n_p – total number of passes in ruler elements; l_0 – cell length; p_0 – maximum number of transit traces passing through the cell of the base crystal; designation [A] defines a non-negative minimal integer $B \geq A$; L – ruler length.

The second condition limits the number n_s trails, crossing the vertical section s,

$$n_s \leq n_{max}, \qquad (2)$$

where n_{max} – the total number of horizontal lines in all channels. Condition, as in (2), must be fulfilled for all vertical sections of the crystal.

Optimization of the resulting placement option is performed by the bionspirated method [6, 7] taking into account the criterion of the maximum number of line segments. Moreover, it is enough to control the constraints only in the sections passing through the blocks.

After an improved variant of block placement is found that satisfies the given criterion, the stage of placement of transistor traces is performed. To solve this problem, model T is used in the form of a set of horizontal rows of vertices (the placement of blocks “Fig. 3, a” corresponds to the model shown in “Fig. 3, b”). Each vertex represents an element (in the figure, such vertices are blackened) or an interblock gap and is characterized by a bandwidth that determines the maximum allowable number of transistor traces passing through this vertex. Number of rows in the model T equal to the number of lines on the crystal.

The throughput of the vertex displaying an element is equal to the total number of passes in the elements of this block. The bandwidth of the vertex corresponding to the interblock spacing depends on the crystal design. Usually, this value is determined by the distance between adjacent macrocells, and in a line crystal, the total throughput of interblock blocks is limited by the difference between the long line and the total length of blocks in the line.

For each transistor trace, two lines are known i, j elements in which it connects. Moreover, the line may contain several elements incident to this track. It is required to determine for each transit route the elements and the inter-element gaps through which it passes. Examples of possible configurations of transistor traces are shown in “Fig.2”. The criterion for the optimal placement of transistor traces coincides with the criterion used when placing the elements.

Let in the ruler i vertices incidental to the transistor circuit under consideration lie on the segment [a, b], and in the ruler j on the segment [c, d] “Fig. 3, a”. If the projections of these segments on the horizontal line intersect, then the transit alignment can consist of a single vertical segment. In “Fig. 4, a” three possible positions: t_1, t_2, t_3 transistor traces are shown with a dashed line. To minimize the number of used horizontal lines, transistor traces are divided into two groups: with intersecting projections of the segments [a, b] and [c, d] and with non-intersecting ones. First, for each trace of the first group, a line segment is sought ti. In this case, the following situations are possible: 1) there is no desired segment; 2) there is only one segment; 3) there are multiple valid segments.

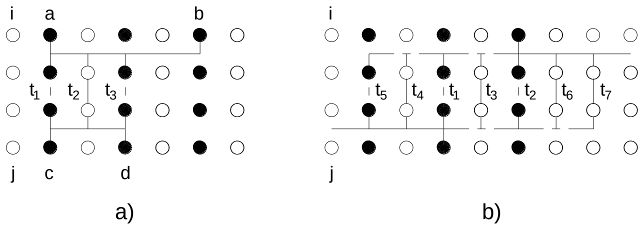

Fig. 3. – Examples of positions of transistor traces

In the first situation, the track is transferred to the second group, in the third, the best segment is selected. In this case, preference is given to the segment that passes through the blocks. Pulling the route through the inter-block gap may require additional area equal to the area of one cell. In addition, such routing may require block displacement and lead to a decrease in the number of LANs. Let us denote by p_ϑ number of transistor traces through the vertex ϑ. If all valid segments pass through vertices with a value p, not a multiple of the cell bandwidth. In addition, preference is given to the segment passing through the vertices that lie on the same vertical line with the vertices ϑ, such, that $p_\vartheta \neq 0$.

For each trace of the second group, there are two most preferred configurations, examples t_1, t_2 which are shown in “Fig. 5, b”. Trails (e.g. t_3), location between t_1 and t_2, also have a minimum length, but one turn more, while tracks located to the left or to the right may have only one turn, but a longer length (tracks $t_4 - t_7$). The configuration of the route is several valid configurations, the best one is chosen according to the same rules that were applied for the routes of the first group [9, 10].

All unplaced traces are implemented using arbitrary configurations. In this case, a vertex is sequentially selected in each intermediate line through which the trace is drawn, and the rules for choosing the next vertex coincide with the rules for choosing a vertical segment of the trace of the first group. At the next stage, the exact values of the coordinates of the segments and transistor traces on the crystal are determined. This operation is performed using the model T by sequential placement of fragments in the rulers so that the left border of each block coincides with the left border of the cell on the crystal.

The final stage is the placement of elements and transistor traces inside each block. The initial placement is performed by a sequential algorithm according to the criterion coinciding with that used in the previous stages. To assess the optimality criterion, it is necessary to know the location of the traces connecting the elements in the considered ruler and the elements placed in adjacent rulers [11]. The position of the trace is characterized by a horizontal line interval, the left boundary of which coincides with the minimum coordinate of the output incident to this target, and the right one - with the maximum. During the placement of elements, the spacing of the chains is refined. Before placing elements, the spacing of the nets is determined by the coordinates of the centers of the blocks. So, chains 1 and 2 characterize the position of already placed elements. Simultaneously with the elements, the transistor traces defined at the previous stage are also placed inside the block, which are assigned to specific passes in the elements.

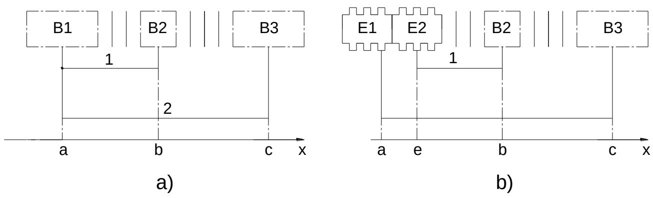

Fig. 4. – Examples of positions of transistor traces

After the initial placement, a genetic algorithm is used with the help of which the pairs of adjacent block elements are permuted. Consideration of pairs of only neighboring elements leads to simplification of the determination of the criterion change. The peculiarity of the algorithm is that when determining the best arrangement of neighboring elements, the best assignment of transistor traces to the passes of these elements is selected. If an element contains logically invariant leads or groups of such leads, then an additional optimization procedure can be applied, during which the corresponding nets are assigned to invariant leads.

III. HYBRID GENETIC ALGORITHM

When modeling, it is possible to combine all types and forms of evolution. Figure 5 shows the basic structure of genetic search based on the use of evolutionary models of Darwin, Lamarck, de Vries. Here, on the basis of the scale of evolution, when interacting with the external environment, signals are generated 0 - for the implementation of Darwin's evolution; 1 - for the implementation of the evolution of Lamarck; 0.5 - for the implementation of de Vries.

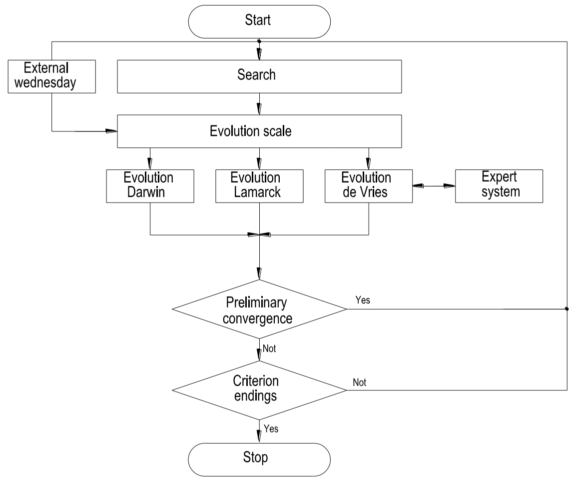

Fig. 5. – Basic structure of genetic search based on the evolution of Darwin, Lamarck and de Vries.

The expert system allows you to formulate a set of rules for constructing the objective function and the end of the search. Repetition of the evolutionary search is possible for preliminary convergence of the algorithm or upon reaching a given value of the objective function. A feature of this scheme is the use of search strategies.

Figure 6 shows a modification of the basic structure of genetic search based on the use of the evolution models of Darwin, Lamarck, de Vries and Popper. Here, in contrast to the basic structure, the scale of evolution, interacting only with the external environment, generates signals to choose the evolution of Darwin (0), Lamarck (1), de Vries (0.5). After that, the Popper evolution model is executed, which implements one of the types of heuristic search in the form of a trial and error method.

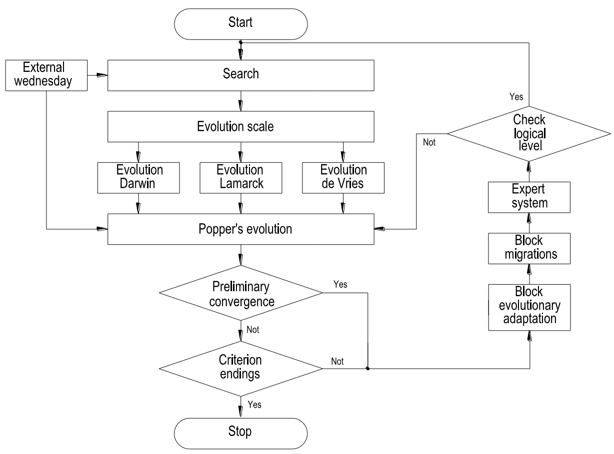

Fig. 6. – Modification of the basic structure

Adaptation and migration blocks have been added to the feedback loop. Adaptation and migration blocks make it possible to build order out of chaos, establish a balance in the system, and select parameters to control evolutionary search in order to obtain optimal and quasi-optimal solutions.

In terms of the number of traces realized and elements placed, the modified genetic algorithm, on average, gave a result that is 17-23% more efficient than the basic structure of genetic search based on the evolution of Darwin, Lamarck and de Vries. An example of optimal placement by a modified genetic algorithm is shown in “Fig. 7”.

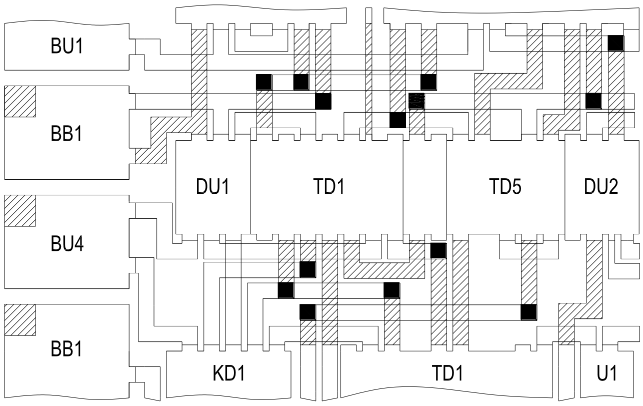

Fig. 7. – Fragment of the scheme of optimal placement of elements by a modified genetic algorithm

When placing elements on a crystal model using a modified genetic algorithm, the placement of the placed elements has an optimal shape, which gives an advantage at the stage of correcting the circuit topology.

IV. Conclusion

The article describes a new approach to solving the problem of placing elements on a chip based on GA, which makes it possible to create an algorithmic environment in the field of genetic search for solving the problem of placing elements on a crystal, taking into account the criterion for the maximum number of line segments and possible restrictions. The criterion of linear segments for the implementation of the technique and possible limitations during the initial placement are described. Examples of the positions of the laid routes, the structure of the modified basic structure of the genetic algorithm are given. A comparative test was carried out with the basic structure of the genetic search, which showed the advantages of the modified genetic algorithm.

Acknowledgment

The reported study was funded by RFBR, project number № 20-37-90151.

The study is supported by the Innovation Assistance Fund under the agreement number 437GUTSES8-D3/62058 dated October 05, 2020.

References

[1] V.I. Danilchenko and V.M. Kureichik, “Genetic algorithm for planning the placement of VLSI,” Izvestiya SFU, vol. 2., 2019, pp. 75-79.

[2] V.V. Kureichik and Vl. Vl. Kureichik, “Integrated algorithm for placing VLSI fragments,” Izvestiya SFU. vol. 7., 2014, pp. 84-93.

[3] V.V. Kureichik and Vl. Vl. Kureichik, “ Bio-inspired research in design and management,” Izvestiya SFU. vol. 11., 2012, pp. 178-183.

[4] A.A. Kalentiev, D.V. Garais, I.M. Dobush and L.I. Babak, “Structural and parametric synthesis of microwave transistor amplifiers based on a genetic algorithm using models of monolithic elements,” TUSUR Reports, No. 2 (26), Part 2, December 2012, pp. 104-112.

[5] G. Moshits, “ Designing active filters,” G. Moshits, P. Horn. – M.: World, 1984. – 320 c.

[6] A.E. Goryainov, “Construction of parametric models of passive components of microwave monolithic integrated circuits using the Extraction-P program,” A.E. Goryainov, I.M. Dobush, L.I. Babak, Nast. Sat. Pp. 94–99.

[7] A.V. Frolov, V.V. Lanovenko and V.A. Chekalov, “Calculation of active filters,” study allowance, Komsomolsk-on-Amur: KnAGTU, p. – 131, 2011.

[8] J.R. Koza, “Genetic Programming II Automatic Discovery of Reusable Programs,” A Bradford Book The MIT Press Cambridge, Massachusetts London, England 746 s.

[9] A.A. Kokolov, I.M. Dobush, F.I. Sheerman and L.I. Babak, et al., “Complex functional blocks of broadband radio frequency amplifiers for single-chip L- and S-band receivers based on SiGe technology,” 3rd Intern. scientific. conf. Electronic Components and Electronic Modules (International Forum - Microelectronics-2017), Alushta, 2017. - M .: Technosphere, 2017, - pp. 395-401.

[10] D.E. Bocklemann and W.R. Eisenstadt, “Combined Diff erential and Common-Mode Scattering Parameters,” Theory and Simulation. IEEE Trans. on Microwave Theory and Techniques, Vol. MTT-43, No. 7, pp. 520–523, July 1995.

[11] T. Sripramong and C. Toumazou, The Invention of CMOS Amplifiers Using Genetic Programming and Current-Flow Analisis, IEEE Trans. CAD, 2002.

[12] V.V. Kureichik and D.Yu. Zaporozhets, “Application of the swarm algorithm for solving the problem of placing VLSI elements,” Informatics, computer technology and engineering education, 2010, № 1. pp. 13-18.

[13] A.A. Kalentyev, D.V. Garays and L.I. Babak, “Genetic-Algorithm-Based Synthesis of Low-Noise Amplifi ers with Automatic Selection of Active Elements and DC Biases,” European Microwave Week 2014, Rome, Italy, pp. 520–523, October 2014.

[14] L.I. Babak, A.A. Kokolov and A.A. Kalentyev, “A New Genetic-Algorithm-Based Technique for Low Noise Amplifier Synthesis,” European Microwave Week 2012, Amsterdam, The Netherlands, pp. 520–523, November 2012.

[15] J.D. Hadley and B.L. Hutchings, “Designing a partially reconfigured system,” in Field Programmable Gate Arrays (FPGAs) for Fast Board Development and Reconfigurable Computing, John Schewel, Editor, Proc. SPIE 2607, 1995, pp. 210-220.

[16] T. Higuchi, T. Niwa, T. Tanaka, H. Iba, H. de Garis and T. Furuya, “Evolvable Hardware with Genetic Learning,” Proc. Sinmlation of Adaptive Behavior, MIT Press, 1993, pp. 417-424.

[17] V.I. Danilchenko, E.V. Danilchenko and V.M. Kureichik, Software implementation of the hybrid algorithm for the placement of VLSI elements using a modified genetic algorithm, computer program: copyright holder of the Southern Federal University, p. 5, 2020.

[18] M. Dorigo , V. Maniezzo and A. Colorni, “The Ant System: Optimization by a colony of cooperating objects,” IEEE Trans. on Systems, Man, and Cybernetics. Part B. N 26(1), 1996, pp. 29-41.

[19] I.P. Norenkov, Basics of computer-aided design. - M .: Publishing house of MSTU named after N.E. Bauman, p. 360, 2000.

[20] S.D. Scott, A. Samal and S. Seth, A Hardware-Based Genetic Algorithm, Dept. of Computer Science Washington University St. Louis, MO 63130-4899, p. 7, 2001.