Modified swarm algorithm «ant tree» in the problem of diversification of tractor resources spaces

Boris K Lebedev¹, Oleg B Lebedev¹, Olga A Purchina²

¹Southern Federal University, Rostov-on-Don, Russia
²Don State Technical University, Rostov-on-Don, Russia

E-mail: lebedev.b.k@gmail.com

Abstract. In this work, the problem of increasing traceability is solved by diversifying trace resources, which consists in using additional areas adjacent to the channels below and above, with a trace layer located above the “over-the-cell” cells. For a single-layer routing uses a new paradigm of combinatorial optimization – ant tree (trees ant colony optimization (T-ACO)), based on the ideas of adaptive behavior of ant colony. The decision tree is used as a decision search graph. An agent for the decision search graph does not create a route, but a tree, which in its structure coincides with the representation of the solution to the trace problem in the «over-the-cell» region. This eliminates the use of additional transformations in the decoding process of decisions that allow interpretation of decisions in the form of trees and allows you to discard a lot of «illegal» solutions, which leads to an increase in the quality of the solutions obtained. Using the tracing procedure in «over-the-cell» based on the ant algorithm allows you to unload the channel by 15-20%. The time complexity of the algorithm depends on the number of vertices n of the decision search graph, the number of agents y, the number of iterations l, and is defined as $O(n^2 \cdot l \cdot y)$.

1. Introduction
One of the most important tasks that are solved when designing the VLSI topology is channel routing. A channel is an area bounded by two lines of contacts: upper and lower [1]. A reference orthogonal network is applied to the trace area, along the lines of which the paths pass. Horizontal lines are called highways. Vertical lines pass through the pins. In channel routing each circuit connecting equipotential conclusions is represented as a set of horizontal and vertical fragments (sections). In the general case, the channel routing problem (CRP) is considered as the problem of locating a plurality of horizontal sections on a plurality of highways subject to restrictions. There are many tracers, some of which allow obtaining almost optimal solutions [1-5]. However, the problem of minimizing the channel trace area is still relevant. A shortage of the trace resource often leads to the fact that during the detailed tracing, the conductors are laid bypassing the loaded areas, which leads to an increase in the delays in the connections and to a decrease in the speed of the VLSI in general. Moreover, a shortage of the trace resource with insufficiently high-quality global trace may lead to the impossibility of 100% detailed trace.

In this work, the problem of increasing traceability is solved by diversifying trace resources, which consists in using additional areas adjacent to the channels below and above, with a trace layer located above the “over-the-cell” cells, as shown in Figure 1. Then some of the connections can be redirected from the channel to the subcellular regions (over-the-cell), which will lead to a reduction in the
number of highways that are used in the channel [4-8]. Redirected connections are traced in a single layer. The goal of a subcellular tracer (ST) is to create the maximum planar sketch of the topology in the (over-the-cell) areas, this should allow minimizing the channel density [4,5].

Figure 1. The structure of the switching field.

Among the existing trace algorithms in the subcellular region [2], the most common are: genetic algorithms (GA) (authors: B.M. Goni, T. Arslan, B. Torton); an algorithm based on dynamic programming methods (DPM) (authors: N. Holmes, N. Sherwani, M. Sarrafzadeh). Due to the fact that the complexity and dimension of the tasks being solved have increased sharply in our time, DPM is rarely used, because they are very laborious. In turn, for the same reasons, the complexity of decoding and the problems associated with obtaining “legal” solutions have sharply increased in genetic algorithms (GA). In addition, there are problems of tracing in areas with broken boundaries, therefore, the development of a tracing algorithm in the areas, which could meet modern requirements, is an urgent scientific task.

Basic procedures for solving the task of over-the-cell tracing.
1. The breaking of multi-terminal circuits into two-terminal connections (TTC).
2. Formation of sets of two-terminal connections – candidates for placement in the subcellular region.
3. Formation of a directed graph of the search for solutions \( G \) in the form of a tree defining a binary embedding relation between each pore of vertices corresponding to a TTC.
4. Formation of a complete graph of \( R \) intersections, reflecting the presence of an unavoidable intersection between each pair of TTC.
5. The selection in the graph \( G \) of the oriented decision tree \( T=(V_u, U_w) \), such that between each pair of vertices of the TTC there is no irreducible intersection ratio.

First of all, the formation of the set \( F=\{f_i|i=1,2,...,m\} \) of two-terminal connections is being made, which are candidates for transfer from the channel to the over-the-cell. Each of the multi-pin circuits shown in Figure 2, on each side of the channel, it is divided into two-output chains (fragments) that connect the contacts in the line.

Let \( l_i \) denote the left edge of the fragment \( f_i \), then \( r_i \) will be right, and \( x(l_i) \) and \( x(r_i) \) will be the coordinates of the edges of the fragment \( f_i \) along the \( x \) axis, which propagates along the line of contacts. If \( x(l_j)<x(l_i)<x(r_j)<x(r_i) \), then between the fragments \( f_i \) and \( f_j \) there is an unavoidable intersection (Figure 3). A graph of unavoidable intersections \( CR=(V, E) \) is formed, the vertex \( v_i \in V \) corresponds to the fragment \( f_i \), i.e. \( f_i=\Gamma(v_i) \), the vertices \( v_i \) and \( v_j \) are connected by the edge \( e_k=(v_i, v_j) \) when \( f_j \) intersects \( f_i \).

We assume that \( f_i \) is embedded in \( f_j \) if:
- \( x(l_j)<x(l_i), x(r_j)>x(r_i) \);
- \( f_i \) and \( f_j \) are located in the subcellular region without intersections, i.e. \( f_j \) will be higher than \( f_i \) (Figure 4).
If \( x(r_i) < x(l_j) \), then the fragments \( f_i \) and \( f_j \) do not intersect and can be located in the subcellular region without mutual intersection. We construct a directed graph of the search for solutions \( G=(V,U) \), the vertex \( v_i \in V \) corresponds to the fragment \( f_i \) (\( f_i = G(v_i) \)). The vertices \( v_i \) and \( v_j \) are connected by an edge directed from \( v_i \) to \( v_j \) only when \( f_j \) is embedded in \( f_i \). The vertex \( v_r \) corresponding to the upper boundary of the over-the-cell is included in \( V \). The vertex \( v_r \) will be associated with each vertex \( v \notin V \), because any of the fragments is embedded in the over-the-cell, which is bounded by the upper boundary. In Figure 5 shows a diagram, and in Figure 6 corresponding to this scheme decision search graph (DSG) \( G \).

Among the set of fragments \( F_i \) embedded in \( f_i \), there is a subset \( F_{is} \subset F_i \) such that no two fragments \( f_k \in F_{is} \) and \( f_j \in F_{is} \) intersect. We call this subset \( F_{is} \) the \( s \)-th embedding of the fragment \( f_i \), and the subset \( B_{is} \) the \( s \)-th embedding of the vertex \( v_i \). There can be several sets such as \( F_{is} \cup F_{is} = F_i \).

2. Statement of the problem

There is a set \( F = \{f_i\} \) of two-terminal connections (fragments), which are candidates for transfer from the channel to the region. Based on the set \( F \), an oriented decision search graph \( G=(V,U) \) is constructed in the form of a tree, which reflects the binary embedding ratio and the complete graph of unrecoverable intersections \( CR=(V,E) \). Each of \( f_i \) fragments corresponds to the cost \( c_i \). It is necessary to distinguish the subgraph \( T_w = (V_w, U_w) \), \( T_w \subset G \), in DSG \( G \), which has the properties [6, 9]:

- \( T_w \) is a tree with a root vertex \( v_r \);
- \( B_{iw} \) is a set of child vertices of \( v_i \in V_w \), \( B_{iw} \subset V_w \); for any \( v_i \in V_w \), the set \( B_{iw} \) is an embedding of the vertex \( v_i \), in other words, fragments of the set \( F_{iw} = G(B_{iw}) \) are disjoint and embedded in the fragment \( f_i \).

The \( T_w \subset G \) tree with the properties listed above, in fact, determines the method of planar laying in the subcellular region of fragments that correspond to the vertices of the set \( V_e \subset V \). We will call this subgraph the decision tree (DT). Figure 7 shows a variant of DT \( T_j \) with planar stackings of fragments that correspond to the vertices of the DT.
Suppose that each of \( f_i \) fragments corresponds to the cost \( c_i \), then the decisive tree \( T_w=(V_w,U_w) \) corresponds to the total cost: \( \psi_w=\sum\{c_i\}, i|v_i\in V_w \).

The parameter \( \psi_w \) is taken as an optimization criterion. The goal of optimization is to maximize the parameter \( \psi_w \). We recursively determine the level \( \alpha_{w_i} \) of the fragment \( f_i \) (vertices \( v_i \)) that is part of the decision tree \( T_w \) [9]. The vertex \( v_i\in V_w \), which does not contain vertices embedded in it, has a zero level (\( \alpha_{w_i}=0 \)). Let \( \mu_{w_i} \) be the maximum level among the vertices embedded in \( v_i \), therefore, the level \( v_i \) equals \( \mu_{w_i}+1 \), \( \alpha_{w_i}=\mu_{w_i}+1 \). Note that the vertex \( v_i \) (fragment \( f_i \)) in different decision trees can have a different level \( \alpha \). We call the \( t_{w_i} \) block the subtree of the decision tree \( T_w \), whose root has the vertex \( v_i\in V_w \). The level of the \( t_{w_i} \) block is equal to the level of the vertex \( v_i\in V_n \), which is included in the decision tree \( T_w \). We assume that the \( t_{w_i} \) block is part of the \( t_{w_j} \) block if, in the decision tree \( T_w \), the root vertex \( v_i \) of the \( t_{w_i} \) block is a child vertex of the root vertex \( v_j \) of the \( t_{w_j} \) block and \( \alpha_{w_j} \geq \alpha_{w_i}+1 \). Blocks \( t_{w_i} \) and \( t_{w_j} \) will be considered disjoint only if \( v_i \) and \( v_j \) do not intersect with each other. Let the set of vertices \( B_{w_i} \), which are the embeddings of the vertex \( v_i \), correspond to the set of blocks \( Q_{w_i} \). We establish the value \( \omega_{w_i} \) of the \( t_{w_i} \) block – the cost \( \omega_{w_i} \) of the \( t_{w_i} \) block located at a non-zero level (\( \alpha_{w_i} \neq 0 \)) will be equivalent to the total cost of the set of blocks \( Q_{w_i} \) that are part of the \( t_{w_i} \) block and the cost \( \phi_i \) of the vertex \( v_i \).

\[
\omega_{w_i}=c_i+\sum\{\omega_{w_j}\}, j|v_j\in B_{w_i}.
\]

The vertices \( v_i\in V_w \), located at the zero level, play the role of a zero-level block, its value is equal to the value of \( v_i \); \( \omega_{w_i}=c_i \). Therefore, \( \psi_w=\omega_{w_r} \) – is the cost of the block \( t_{w_r} \), with the root at the vertex \( v_r \), equal to the cost of the DT \( T_w \). In turn, \( t_{w_r} \) and \( T_w \) will correspond to each other. To set the number of \( m_w \) lines used for laying the DT – \( T_w \), we use the value of the level \( \alpha_{w_r} \) of the root vertex (\( m_w=\alpha_{w_r} \)).

3. Single-layer tracing in the subcellular region based on combinatorial optimization of “ant tree”

In [6-10], the researchers proposed a description and analyzed the existing single-layer trace algorithms. To build the most effective modern algorithm that meets all the requirements, resort to the use of various innovative approaches and technologies. Now one of the scientific areas is being actively developed, which includes various mathematical methods, the construction of which is carried out directly on the basis of the principles of natural decision-making mechanisms [11]. To create a decisive tree, a combinatorial optimization structure is used. It is called “trees ant colony optimization (T-ACO).” This structure, in turn, is created on the ideas of adaptive behavior of an ant colony [12,13-15] and provides an opportunity to perform tree synthesis [14]. The optimization task, having the form of a T-ASO structure, is as follows:

- building a decision search graph (DSG) in the form of a tree;
- the creation of a swarm of agents of feasible decision trees on the DSG;
- choosing the best solution.

The task of the swarm of agents is to construct on the graph \( G=(V,U) \), by sequentially choosing the set of vertices \( V_w\subseteq V \), a route \( M_w \) with the following properties. The route \( M_w \) is an ordered sequence of sub-routes \( M_w=\{M_{w_i}|i=1,2,...,m\}; U(M_w)=M_w \). The subroute \( M_{w_i} \) is an ordered sequence of the set of mutually disjoint vertices \( V_{w_i}\subseteq V_w \), \( U(V_{w_i})=V_w \). The vertices of the set \( V_{w_i} \) correspond to the set of
fragments placed on the \(i\)-th line of the SR. The number of sub-routes corresponds to the number of highways in the SR. The subgraph of the graph \(G=(V,U)\), including the set of vertices \(V_u \subset V\), is actually the decisive tree \(T_u=(V_u,U_u)\), which defines the method of planar stacking of fragments in the subcellular region.

As a decision search graph, the authors opted for a modified solution search graph \(G=(V,U)\). The main difference between the modernized solution search graph is that there are no edges in the graph that correspond to the closure of transitive embedding routes. If \(v_i\) is embedded in \(v_j\), then in \(G\) there is only one route of unit length connecting the edge \(v_i\) and \(v_j\) directed from \(v_i\) to \(v_j\). The development of the decision graph starts from the main vertex \(v_r\). This vertex, in turn, corresponds to the upper point of the subcellular zone [14]. Initially, taking into account the results obtained, during the analysis of the circuit, the construction of the set of vertices \(V_j\) is performed. Any of these vertices can be part of only the main vertex \(v_r\), in our example (Figure 8) \(V_7=\{v6,v8,v7,v9,v10\}\). \(v_r\) is connected by edges to all vertices of the set \(V_j\). Then the formation of the set is made. Any of the vertices of the set \(V_j\) can be embedded only in the composition of some vertices of \(V_j\). In other words, any vertex of the set \(V_j\) can be a child vertex of only some vertices of \(V_j\). Between the pairs \(v_i, v_j\) and \(v_j, v_k\), edges are formed that have a direction from \(v_j\) to \(v_k\) if \(f_j\) is embedded in \(f_k\). At step \(n\), taking into account the results obtained, during the analysis of the circuit, the construction of the set of vertices \(V_u\) is performed. Any vertex of the set \(V_u\) can be embedded only in some vertices of the set \(V_u\). Such a procedure is carried out until all fragments of the scheme are displayed in column \(G\). The decision search graph created as an example is shown in Figure 8. Thus, the edges connect the vertices of two adjacent levels.

![Figure 8. Graph for finding solutions.](image)

The result of the ant tree algorithm (T-ACO) is the decision tree \(T_r\) with the best estimate on the decision search graph \(G\). The search for a solution (decision tree \(T_r\)) is reduced to constructing a route with the above properties based on the decision graph \(G\), which has a fixed structure, using an auxiliary \(GT\) decision graph with transforming structure. The initial structure of the graph \(GT\) is identical to the structure of the graph \(G\). As the initial data, the intersection graph \(CR=(V,E)\) is formed between the pairs of vertices \(v_i, v_j\) and \(v_j, v_k\), an edge is created if the fragments \(f_i\) and \(f_j\) intersect.

The search for solutions to the problem is carried out by a team of agents \(A=\{a_k\}_{k=1,2,...,n_k}\). Agent behavior models are associated with the distribution of pheromone at the vertices of graph \(G\), used as a repository of collective evolutionary memory. Initially, at all vertices of the graph \(G\), an equal, small amount of pheromone \(Q_v\) is deposited, here \(v=|U|\), the parameter \(Q\) is set a priori. Finding solutions is an iterative process. At each iteration of the ant tree algorithm, each agent \(a_k\) forms its solution to the problem on the transforming decision graph \(GT\), using the graph \(G\) as the storage of collective memory. The interpretation of the solution is the decision tree \(T_r\subset G\), which will correspond to the sketch of planar tracing in the SR. Iteration \(l\) consists of three stages.

The first stage of each iteration is to find each agent in the population of its solution (route \(M_l\) and tree \(T_l\)), by sequentially transforming the initial auxiliary decision graph \(GT=(V,U)\). The second stage – agents lay the pheromone at the vertices of the decision search graph \(G\). The third stage consists in the evaporation of the pheromone at the vertices of the decision search graph \(G\). The cyclic ant-cycle
method of the ant system is used in this work, while the pheromone is deposited by agents at the vertices of the solution graph $G$ after of how all decisions of the population will be formed. Finding a solution is carried out in the process of sequentially viewing the vertices of the decision tree. Formation by the agent $a_k$ of the route $M_k$ and the corresponding decision tree $T_k$ is carried out by sequential distribution of the vertices $V$ of the graph $G$ along the highways starting from the first. Numbering of highways from bottom to top. After choosing the next highway, many $SK$ peaks are formed to place on it – candidates. When filling the first highway, the set of candidates $SK_1$ includes only child vertices of the graph $GT(0)$. For the DSG (Figure 7), $SK_1=\{v_{11},v_{12},v_1,v_2,v_3,v_4,v_5,v_{10}\}$.

Using the RAZM procedure, the agent $a_k \in A$ forms the maximum cost subset $SK_\gamma$ of mutually disjoint vertices of the set $SK_t$ for placement on the $1$st highway. $SK_\gamma=\{v_{11}, v_1, v_2, v_3\}$. On the intersection graph $CR=(V,E)$, the set $D_1$ of vertices intersecting the vertices of the set $SK_\gamma$ is determined. For our example, $D_1=\{v_{12},v_5,v_6,v_7\}$. After that, the graph $GT(0)$ is transformed into the graph $GT(1)$ by removing from $GT(0)$ the subsets of the vertices $D_1$ and $SK_\gamma$ with all edges of the graph $GT(0)$ incident to these vertices. Further, in the graph $GT(1)$, a set of vertices $SK_2$ is formed, which includes only daughter vertices of the graph $GT(0)$ and the transition to filling line $2$, $\gamma=2$. For our example, $SK_2=\{v_6,v_7\}$.

When filling the highway $\gamma$, the set of candidates $SK_\gamma$ includes only daughter vertices of the graph $GT(\gamma-1)$. Using the RAZM procedure, the agent forms the maximum cost subset $SK_\gamma$ of mutually disjoint vertices of the set $SK_t$ for placement on the $\gamma$th highway. On the intersection graph $CR=(V,E)$, the set of vertices $D_\gamma$ that intersect with the vertices of the set $SK_\gamma$ is determined. Further, the graph $GT(\gamma-1)$ is transformed into the graph $GT(\gamma)$ by removing from $GT(\gamma-1)$ the subsets of the vertices $D_\gamma$ and $SK_\gamma$ with all edges of the graph $GT(\gamma-1)$ incident to these vertices. A set of vertices $SK_{\gamma+1}$ is formed and the transition to the formation of the level $\gamma=\gamma+1$. Agent $a_k$ distributes vertices along the highways after the transformations make the graph $GT(\gamma)$ empty, $GT(\gamma)=\emptyset$, and therefore $SK_{\gamma+1}=\emptyset$. Denote by $V_\gamma(l)$ the set of vertices of the graph $G$ distributed by the agent $a_k$ along the $\gamma$ trunks on iteration $l$. The decision tree $T_\gamma(l)\subset G$ is formed as follows. The set of vertices $V_\gamma(l)\subset G$ of $G$ are labeled. Then, all unlabeled vertices with all edges of the graph $G$ incident to these vertices are deleted from the graph. The remaining part of the graph $G$ is a decision tree $T_\gamma(l)\subset G$. After that, the total $\psi_\gamma(l)$ of the set of vertices $V_\gamma(l)$ distributed by the agent along the lines is calculated, which is an estimate of the solution.

Consider the RAZM procedure. We arrange the set of vertexes $SK_t$ of the candidates for placement on the highway by increasing the coordinates of the left ends of the fragments corresponding to these vertices. We construct the vector $P$. Let $P(v_i)$ be the sequence number $v_i$ in the vector $P$. Then $\langle ij \rangle=\{x(l)\times x(l)\rightarrow P(v_i)\times P(v_j)\}$. We arrange the vertices of the set $SK_t$ in a ruler in the order in which the corresponding fragments are located in the vector $P$. The formation of the maximum cost-effective subset $SK_{\gamma}\subset SK_t$ of mutually disjoint vertices of the set $SK_t$ for placement at the $\gamma$-th level is carried out by the agent $a_k$. A sequentially (step-by-step) by sequentially viewing the elements of the ordered list $SK_t$. At step $t$, the number of the filled line $- \gamma$, the set $SK_\gamma$, the list $SK_t(t)$ of vertices already included in the level being formed, $v_i$ – the last vertex (fragment $f_i$) included in the list $SK_t(t)$ is stored in the agent $a_k$ memory. The agent scans the set of all vertices $SK(\gamma)\subset SK_t(t)$ that are free at this step and are located in the list $SK(\gamma)$ to the right of $v_i$ and selects from it a subset of vertices $Z_t(t)$ $Z_t(t)=SK_t(t)\subset SK_t(t)$ that satisfy the conditions for placing fragments in the current highway:

a) each vertex $v_i \in Z_t(t)$ can be placed on the current filled highway without intersections with the vertices of the set $SK_\gamma(t)$ already placed;

b) if $f_j \in Z_t(t)$ is placed on the highway, then there is no fragment $f_j \in Z_t(t)$ that can be placed in the remaining free zone of the filled highway between the fragments $f_i$ and $f_j$ without violating the restrictions and conflicts (intersections) with $f_j$.

Compliance with the considered conditions is explained as follows. Since the filling of highways with fragments is carried out sequentially using the "irrevocable" strategy, it makes no sense to leave
on the highway not filled (empty) areas into which some fragments can be placed. For each vertex \( f_i \in Z_l(t) \) that meets the placement conditions, three parameters \( \varphi, \lambda, e \) are calculated:

- \( \varphi \) is the total amount of pheromone accumulated at the vertex \( v_i \);
- \( \lambda_i \) is the distance between the end of the last fragment \( f_i \) in the filled line and the beginning of the fragment \( f_i \) placed in one line. If \( f_i \) is located at the beginning of the highway, then \( d_i \) is the distance between the beginning of the highway and the beginning of \( f_i \);
- \( e_i \) is the cost of the fragment \( f_i \).

The integral cost \( Q_\gamma(t) \) of the connection of the vertex \( f_i \in SK_\gamma(t) \) with \( f_i \in SK_\gamma(t) \) is determined by multiplicative convolution using formula (1), and with additive convolution using formula (2).

\[
Q_\gamma(t) = (\varphi_i(l-1) \cdot e_i^\gamma)/\left(\lambda_i^\gamma + 1\right) \\
Q_\gamma(t) = a\varphi_i(l-1) + (\delta/(\lambda_i + 1)) + \beta e_i,
\]

where \( \alpha, \beta, \delta \) are control parameters that are selected experimentally.

The probability \( P_\gamma(t) \) of the inclusion of the vertex \( f_i \in Z_l(t) \) in the formed route \( SK_\gamma(t) \) is determined by the following relation:

\[
P_\gamma(t) = Q_\gamma(t) \theta, \quad \text{where} \quad \theta = \sum_j Q_\gamma_j(t), \quad \delta(f_i \in Z_l(t)).
\]

Consider the heuristic considerations underlying the formulas (1) and (2). The larger \( \varphi_i(l-1) \), the greater the likelihood that the vertex \( f_i \) is part of the optimal route. The smaller \( \lambda_i \), the higher the filling density of the line, which helps to reduce the number of lines required to place fragments. The inclusion of the \( e_i \) parameter in expressions (1) and (2) helps maximize the cost of the subset \( SK_\gamma \subset SK_\gamma \) of mutually disjoint vertices of the set \( SK \) for placement on the \( \gamma \)-th trunk, which indirectly improves the search efficiency and minimizes the target estimate — the number of highways. An agent with probability \( P_\gamma(t) \) selects one of the vertices \( f_i \in Z_l(t) \), which is located on the \( \gamma \)-th line. After this, the set of placed vertices is already denoted as \( SK_\gamma(t+1) \), and \( f_i \) is excluded from the set \( Z_l(t) \). For \( \alpha = 0 \), the selection of the \( f_i \) fragment closest to \( f_i \) is most probable, that is, the algorithm becomes similar to the “left end” algorithm. At \( \beta = 0 \), the probability of selection practically depends only on the amount of pheromone on the edge, which leads to suboptimal solutions. The relationship between these values is found experimentally. After the formation of decision trees by all agents, pheromone is deposited on the vertices of the DSG. Each agent lays on the vertices \( V_d(l) \subset V \) pheromone in the amount \( \tau_d(l) \) proportional to the estimate \( \psi_d(l) \) of the decision tree:

\[
\tau_d(l) = c \psi_d(l),
\]

where \( c \) is the coefficient.

We denote by \( \varphi_d(l) \) the total amount of pheromone accumulated at the vertex \( v_i \in V \) in the DSG \( G \) after the agents completed the second stage of the \( l \)-th iteration.

At the third stage of iteration, at the vertices of the graph \( R_f(F,E) \), in accordance with formula (5), the pheromone evaporates.

\[
\tau_d(l) = \tau_d(l) \cdot (1 - \rho),
\]

where \( \rho \) is the update coefficient.

The last steps in the iteration are related to finding the best solution that is remembered. After that, the next iteration is performed. The time complexity of this algorithm depends on the number of vertices of the graph \( n \), the number of ants \( y \), the number of iterations \( l \), and is defined as \( O(n^{2.1} \gamma) \).

4. Experimental studies

The aim of the experiment was to determine the number of iterations of the ant algorithm, which with a high probability (close to unity) would guarantee the finding of the optimal solution. Initially, research was conducted aimed at finding the values of control parameters that could ensure the greatest efficiency of the work of a subcellular tracer OTC-ACO. The results for the ant algorithm are
obtained with the following quantitative values: agents – 100; iterations – 120; the initial deposition of pheromone – 1000; the amount of pheromone deposited by the agent on the solution – 100; pheromone evaporation coefficient – 0.7.

A graph of the complexity of the tracing algorithm in the subcellular region on the number of chains is shown in Figure 9. In all solved examples, optimal results were obtained. The complexity of the algorithm \( O(n^2) \). The dependence of the quality and the best solution in the population on the number of iterations is shown in Figure 10.

![Figure 9](image1.png)

**Figure 9.** The dependence of the complexity of the tracing algorithm of subcellular tracer OTC-ACO in the SR on the number of chains.

![Figure 10](image2.png)

**Figure 10.** Dependence of quality and the best solution in the population on the number of iterations.

Table 1 shows the values for ten test cases obtained by the tracer without using the subcellular region, and the results obtained by the tracer using the developed ant tracing algorithm in the subcellular region. The success of the algorithm depends on the correct control settings. The use of more sophisticated modified strategies for choosing alternatives when laying routes can become a source of improvement in the ant algorithm operation. A series of examples has been synthesized, with the optimal values of the objective functions known in advance, to analyze the accuracy of the solutions obtained using this algorithm.

| Example | Without SR (quantity highways) | With SR (quantity highways) | Channel unloading in % |
|---------|--------------------------------|------------------------------|------------------------|
| 1       | 14                             | 12                           | 14.3%                  |
| 2       | 12                             | 10                           | 16.7%                  |
| 3       | 14                             | 9                            | 35.7%                  |
| 4       | 20                             | 16                           | 20%                    |
| 5       | 24                             | 21                           | 12.5%                  |
| 6       | 25                             | 22                           | 12%                    |
| 7       | 29                             | 24                           | 17.2%                  |
| 8       | 32                             | 26                           | 20.8%                  |
| 9       | 33                             | 26                           | 21.2%                  |
| 10      | 35                             | 28                           | 25%                    |

In this case, the probability of finding optimal solutions is 0.93, and the general estimate of the time complexity is within \( O(n^2) - O(n^3) \).
5. Conclusion
In the work, the problem of increasing traceability is solved by diversifying trace resources, which consists in using additional areas adjacent to the channels from below and above, with the trace layer located above the cells—“over-the-cell”. Then part of the connections can be redirected from the channel to the subcellular region, which will lead to a reduction in the number of highways that are used in the channel. Redirected connections are traced in a single layer. The goal of a subcellular tracer is to create the maximum planar sketch of the topology in the subcellular region, this should allow minimizing the channel density. In order to build a sketch of a single-layer trace, the structure of combinatorial optimization “ant tree” (trees ant colony optimization (T-ACO)) was used. An ordered modified decision graph \(G=(V,U)\) is used as a solution search graph; its construction is carried out by recursively determining the possible embedding of fragments into each other. The difference from the canonical structure of the ant algorithm is that the ant on the decision search graph does not create a route, but a tree, which in its structure coincides with the representation of the solution. This eliminates the use of additional transformations in the process of decoding decisions that allow interpretation of decisions in the form of trees and allows you to discard many “illegal” decisions, which leads to an increase in the quality of the resulting solutions. Using the tracing procedure in the subcellular region allows you to unload the channel by 15-20%. The time complexity of this algorithm depends on the number of vertices of the graph \(n\), the number of agents \(y\), the number of iterations \(l\), and is defined as \(O(n^2 \cdot l \cdot y)\). In this case, the probability of finding optimal solutions is 0.93, and the general estimate of the time complexity is within \(O(n^2)\).

Acknowledgements
This research is supported by grants of the Russian Foundation for Basic Research of the Russian Federation, the project № 18-07-00737.

References
[1] Nemudrov V, Martin G 2004 Systems-on-chip. Design and development (M: Technosphere) p 157
[2] Bozhko A, Karpenko A 2018 Proceedings of the Vth International workshop «Critical infrastructures: Contingency management, Intelligent, Agent-based, Cloud computing and Cyber security» (IWCI 2018) DOI:10.2991/iwci-18.2018.2.
[3] Bozhko A, Rodionov S 2016 Science and Education (M: Publishing House MSTU N.E. Bauman) DOI: 10.7463/0816.0844719
[4] Karpenko A P et al, 2015 Fundamentals of computer-aided design, Textbook (M.: LLC "Scientific and Publishing Center INFRA-M") p 332
[5] Litvinov E I et al 2014 Problems of developing Advanced Micro- and Nanoelectronic Systems. Proceedings (M.: IPPM RAS) p 51
[6] Bykov S O 2014 Problems of developing Advanced Micro- and Nanoelectronic Systems. Proceedings (M.: IPPM RAS) p 21
[7] Kazennov G G 2005 Basics of designing integrated circuits and systems (M.: Binom. Laboratory of Knowledge) p 295
[8] Garbulina T et al 2016 Problems of developing Advanced Micro- and Nanoelectronic Systems Proceedings (M.: IPPM RAS) p 165
[9] Lebedev O et al 2015 Bulletin of the Southern Federal University. Technical science 6 125
[10] Lebedev O et al 2015 Collection of scientific papers of the VIII International Scientific and Practical Conference "Integrated Models and Soft Computing in Artificial Intelligence" 2 624
[11] Karpenko A P 2014 Modern search engine optimization algorithms. Algorithms inspired by nature: a tutorial (M: Publishing House MSTU, N.E. Bauman) p 448
[12] Dorigo M et al 2004 Ant Colony Optimization (MIT Press, Cambridge MA) p 187
[13] Lebedev O B 2013 2013 Models of adaptive behavior of an ant colony in design problems
[14] Lebedev B K et al 2014 Proceedings of the Congress on Intelligent Systems and Information Technologies “IS – IT’14”. Scientific publication in 4 volumes (M.: Publishing house Fizmatlit) p 69

[15] Ryzhenko N et al 2016 Problems of developing Advanced Micro- and Nanoelectronic Systems Proceedings (M.: IPPM RAS) p 150