Algorithms for calculating schemes of transport routes in a felling area

G E Rego*, O I Grigoreva and R V Voronov

1Petrozavodsk State University, 33 Lenin Avenue, Petrozavodsk 185035, Russian Federation
2St. Petersburg State Forest Technical University, 5 Institutskiy Lane, St. Petersburg 194021, Russian Federation

*Corresponding e-mail: regoGr@yandex.ru

Abstract. The study considers the problem of constructing routes for the export of timber. The task reduced to building a rooted tree. The study presents two approximate algorithms for solving the problem of covering polygons with rooted trees. To simplify the task, the polygon is covered with a square grid. A greedy algorithm is described that should give an approximate solution. The article also describes the branch and bound method with a reduced number of iteration options. We have reduced the number of iterations by using a special way of building links.

1. Introduction

The optimality of the transport development of the felling area (the choice of the layout of the skidding trails) directly affects the efficiency of the logging work – productivity, specific fuel consumption, environmental consequences for soil and trees left for growing. With a dense network of skidding trails, their cargo work is reduced, and hence the degree of impact of forest machinery propellers, but the costs of creating and maintaining trails increase. With a small number of trails per unit of cutting area, it becomes more difficult to collect timber from semi-fallen trees, the risk of damage to the trees and undergrowth left behind increases. It should be borne in mind that in practice the cutting area is allocated in the form of a forest inventory unit and almost never has the correct form. Almost all felling areas have microdepressions, which are often key biotopes that cannot be cut, and microhills, which it is desirable to avoid. On felling areas, there are often non-operational areas (seed groups, clumps of undergrowth, glades), through which skid lines should also not be laid. All this makes the task of developing a mathematical apparatus to support the decision-making on the choice of the optimal transport development of the cutting area is extremely urgent and rather difficult.

The rules of timber harvesting and the specifics of timber harvesting in forestries, specified in Article 23 of the Forest Code of the Russian Federation [1], impose restrictions on the area occupied by portages and loading points – it should not exceed 15% of the total area of the cutting area. Usually, the width of the apiary trail is no more than 4 meters. The width of the main road is usually from 5 to 6 m.

Currently, approximately 70% of all timber harvested in Russia, and almost 100% in Northwest Russia, is trucked by forwarders – in a fully submerged position. In the warm season, in order to maintain cross-country ability, they try to make as few trips as possible along the drag, ideally to...
collect all the assortments in one pass. To do this, they prefer to take a heavier (lifting) forwarder and assemble in 1–2 times than to use a light one, which will collect in 4–6 passes.

There is a rule – the wood should not be taken to the forest; collection is made when driving to the loading point. When using forwarders, circuits with looped apiary trails are usually used. When assessing the results of the main logging operations, two main aspects can be distinguished: operational (technological) and environmental efficiency. The first is assessed by such indicators as energy and labor intensity of the process, unit costs (prime cost), etc. [2, 3].

Environmental efficiency will be assessed according to the degree of negative damage to the forest ecosystem, and we note that damage to the forest environment during logging can be positive for subsequent natural reforestation, [2, 4]. New methods for assessing environmental efficiency both for the industry as a whole and for the forest industry separately [5, 6] have been developed recently.

With regard to logging production, it is proposed to consider environmental efficiency as a "component of the vector of overall efficiency", this allows us to assert that the concept of environmental efficiency is inextricably linked with the economic, technical, technological and other parameters of the entire production process as a whole.

In this regard, it is required to develop a methodology for optimizing the placement of routes for the movement of forest machines from the point of view of reducing energy consumption, and, accordingly, the cost of the most expensive technological operation of the main logging operations – skidding of timber.

Developed by the participants of the scientific school "Innovative developments in the field of the logging industry and forestry", the coordinate-volumetric technique for tracing skidding trails, at its core, is based on the principle of dividing the logging area into subsections with the establishment of their centers and laying the trails of skidding trails through them [2-4]. This approach allows us to optimize the performance of the primary forest transport (skidders) both in terms of operational and environmental performance [5–9].

A lot of research is devoted to the problem of covering polygons with flat figures in various interpretations [10]. For example, the NP-complete problem of covering a polygon with rectangles of a given shape is a special case optimization problem about cutting [11]. Maps of land cover can be obtained by remote sensing.

The study considers the problem of covering a polygon with a rooted tree. This problem is a continuation of the previous study, which was devoted to solving the problem of finding the minimal disjoint paths on the polygon [12, 13].

The example of the problem is shown in figure 1.

Figure 1. An example for the problem of transport routes in a felling area.
2. Results and Discussion

2.1. Mathematical model

The plot model is a polygon – a closed polyline on a plane. The polygon may have holes (i.e., have closed broken lines inside) or it may have no holes. The task of constructing a scheme of transport routes is the task of covering the polygon with a forest of root trees. A tree is a connected acyclic graph. This is a graph in which we can get from any vertex to any other, and in only way, moving along edges (links) of the graph. Tree roots are located on the outer border of the polygon. The roots are the sources of the transport network. An important feature of our tree is that the links can be adjacent to each other. This means that a link can start from an inner point of another link in the tree. Covering a polygon with a forest of trees means that the distance from any point of the polygon to the nearest point of the forest does not exceed \( \varepsilon \) for some \( \varepsilon \). We will call such a root forest an-\( \varepsilon \)-covering forest of the polygon. The following initial conditions are possible:

1. The number and position of the roots are any, and the algorithm can locate them arbitrarily on the border of the polygon.
2. The number of roots is arbitrary, but they are located only on certain specified segments of the polygon boundary.
3. The number of roots is fixed, the roots are located arbitrarily on the border of the polygon.
4. The number of roots is fixed and they are located only on specified segments of the polygon boundary. In the degenerate case, there is only one root and its position is fixed.

In this paper, we will consider the 4th case. There will be one root and its position will be fixed. We will use the number of tree links as an optimality criterion. The task is to find an-\( \varepsilon \)-covering tree with the minimum number of links for a given polygon.

The result of solving the problem is an algorithm that builds an-\( \varepsilon \)-covering tree. The input to the algorithm is a polygon \( P \), a root point and a number \( \varepsilon \). As output, the algorithm returns the coordinates of the vertices and a list of edges.

As a solution, we present two algorithms based on approximating a polygon by overlaying a square grid with a square length \( \frac{\varepsilon}{\sqrt{2}} \) (the length of a diagonal of a grid square will be \( \varepsilon \)).

We say that a tree link covers a square if they have a common point. If a link and square have a common point, then an adjacent link can start from any point on the square's border (to avoid additional intersections and not increase the number of links). It makes no sense to start an adjacent link other than on the border of the square for the following reason. Any coverage that can be provided by starting the link from some point that does not lie on the border of the square can also be provided from a point \( b \) lying on the border. For this, it is necessary to cut off from the link all points of the segment \( ab \), except for the point \( b \).

We will call squares, not all points of which are in the polygon area, incomplete squares. Incomplete squares can be covered by covering adjacent squares. Let \( ab \) thefarthest point of such a square, from the nearest link. If the distance \( d \) from this point to \( l \) does not exceed \( \varepsilon \) (\( d \leq \varepsilon \)), then such a square will be considered covered by the link \( l \).

Suppose that the algorithm will be completed in at most \( n \) iterations, and at each iteration the number of possible links does not exceed \( m \). Then \( N = \{1,...,n\} \) is the set of all iterations, \( M = \{1,...,m\} \) is the set of possible numbers of links at each iteration. Let \( c_{ij} \in C \ (i \in N, j \in M) \) be the number of squares that are covered at the \( i \)-th step by the \( j \)-th link.

2.2. The greedy method

Consider a greedy method to the problem. Let \( L \) be the set of all links, and \( l_{ij} \in L \ (i \in N, j \in M) \) be the \( j \)-th link at the \( i \)-th step.

At each iteration, we choose a link \( l_{ij} \) that will cover the maximum number of squares from the uncovered part of the polygon.
Building the trajectory of a specific link looks like this. There is a source square (point \( A \)) and a target square (point \( B \)). Between these squares, there is a finite number of squares \( k \). These \( k \) squares can have \( 2k + 6 \) neighbors (+6 are the neighbors of the source and target squares). Some of these neighbors may be incomplete squares. Each neighbor that is an incomplete square or a pair of neighbors can first be considered separately and find such corridors \( d \in \mathcal{D} \), for which the current trajectory covers either just one of the two neighbors or both of them. Next, we consider various combinations of trees from the set \( D \) and choose the one that covers the largest number of squares.

We will describe an algorithm for obtaining an approximate solution based on a greedy heuristic. The structure of the algorithm is similar to Prim’s method for constructing the minimum spanning tree of an undirected graph. The idea of the algorithm is as follows. Initially, the \( \varepsilon \)-covering tree \( T \) of the polygon consists of a single root vertex, which coincides with the source point. Further, at each step, a link is added to the tree that satisfies the following conditions:

1. All link points are inside the polygon.
2. One end of the link belongs, and the other end does not belong to the current \( \varepsilon \)-covering tree.
3. The link has no interior intersections with the current \( \varepsilon \)-covering tree.
4. The link covers the maximum possible number of grid squares that are not covered by the current \( \varepsilon \)-covering tree.

The steps of the algorithm are presented in more detail below:

Step 0. There is a root point \( r \), from which the construction of the tree begins. The set of squares \( K \) is initialized. \( K \) is split into two subsets \( K_{\text{cov}} \) – the set of covered squares and \( K_{\text{ncov}} \) – the set of uncovered squares. Initially, \( K = K_{\text{cov}} \), and \( K_{\text{ncov}} = \emptyset \). We will assume that \( R \) before the construction of the first link belongs to \( K_{\text{ncov}} \), but at step 1, it is its scope that is built at the first iteration.

Step 1. Construct a visibility area for all edge points of the covered squares.

Step 2. Selecting the link \( l_{ij} \). The link \( l_{ij} \) is chosen so that the number of covered squares is maximum \( c_{ij} \rightarrow \max \). The starting point is chosen on the boundary of the set \( K_{\text{cov}} \) among the squares in which one of the links ends. The link cannot intersect the set \( K_{\text{cov}} \).

Step 3. We need to add the squares covered by the link \( l_{ij} \) to the set \( K_{\text{cov}} \), and remove them from the set \( K_{\text{ncov}} \). If all squares are covered, then END, ELSE go to Step 1.

The presented algorithm can also be applied to a polygon with holes. The result of the algorithm is illustrated in figure 2.

![Figure 2](image-url)

**Figure 2.** An example of covering polygon, where (a) – an example of a polygon with a root point, (b)–the first link, (c)–the area covered by the first link, (d)–the second link, (e)–the area covered by the second link, (f)–the third link, (g) – the area covered by the third link.
2.3. Branch and bound method

To describe the branch and bound method, we need to introduce a set of lower bounds $A$, and $a$ in $A$ is the lower bound for the number of links required to cover $K_{ncov}$. Let us denote by $\Omega$– the set of feasible solutions. Record – the number of links in the best solution at the moment. When initializing the algorithm, the cost of any solution, obtained, for example, by the greedy method, can be taken as the initial record. The optimal solution would be rec $(x^*)$. We will assume that $c$ is the number of links in the solution, and $c_{cur}$ is the number of links at a current step.

Now let’s define a set of possible links $L$. Branching will be done based on this set. We construct all possible combinations of links so that links end in border squares either for a polygon or for the sets $K_{cov}$ and $K_{ncov}$. This will reduce the total number of possible links. Let’s define a set of trees $P$. Every link $l$ selects a subset $p_l$ of the set $P$, which denotes all trees that contain the link $l$. When we consider some subset $p_{cur}$ from $P$ then add all possible links subsets to $P$ and remove the $p_{cur}$. For each subset, we calculate the estimate. At the next iteration, we consider the subset with the lowest estimate.

Let us describe a method for finding estimates for $p$. The covered part of the polygon can divide the uncovered $K_{ncov}$ into two or more parts. Let $G$ be the set of uncovered domains that have no common boundary. $G_k$ is a specific area that has no common boundaries with other areas. The union $G_k$ gives the set $K_{ncov}$. For each isolated section $G_k$, its own estimate $a_k$ is given. The estimate of the number of links to cover the entire polygon is the sum of such estimates $a_k$ in $K$.

Let a rectangle be given with side lengths $s_1$ and $s_2$ minimal. The idea behind finding a rectangle with minimum length or width is as follows:

0. We declare a variable short, which will characterize the required minimum (length or width) and make it equal to infinity (or some large number, less than which there is necessarily a length or width of one of the rectangles). Let $e$ be the side along which the short rectangle is drawn.
1. Let us denote by $E$ the set of sides of $G_1$. Consider one of the sides $E_{cur} \in G_1$. We lay one side of the rectangle along or parallel to it.
2. The second side is built perpendicular to the first.
3. We finish building the rectangle so that it is not redundant. This condition means that if the length or width of the rectangle is reduced, then there is a point $g \in G_1$ and $g \in F$.
4. Let $s_{min}$ be the smallest side of the resulting rectangle. If $s_{min} < \text{Short}$, then $\text{Short} = s_{min}$ and $e = E_{cur}$. If there is a side that has not yet been considered, then go to Step 1. Otherwise END.

After executing the algorithm, we can find an estimate for the coverage $a_1$ for the region $G_1$. It is equal to $\left \lfloor \frac{m}{2e} \right \rfloor$.

Next, we calculate estimates for each area $G$ and sum them up. This estimate must be calculated for each possible link. If the estimate $a + c_{cur} > \text{rec}(x)$, where $c_{cur}$ is the number of links in the current tree, then such a subset is removed from the set of solutions as useless. Next, select the link with the minimum value $a + c_{cur}$. If the tree is built and $c_{cur} < \text{rec}(x)$, then $\text{rec} (x_{cur}) = a + c_{cur}$, where $x_{cur}$ is the current solution. When all subsets are discarded, the problem becomes solved and the current record becomes the best solution found. The algorithm is illustrated in figure3.
3. Results and Discussion
We presented the algorithms for calculating schemes of transport routes in a felling area. For solution the problem of covering polygons with rooted trees we described the greedy method and branch and bound method. The greedy method is approximate, but it works faster and will give acceptable results in most cases. The branch and bound method is accurate, but slow. It can only be used for small polygons. In the future we will consider genetic algorithms for this problem. We applied a square grid coverage to reduce the number of iterations. We are planning to consider genetic algorithms to solve this problem.

![Figure 3. An example of computing estimate.](image)

References

[1] Forest Code of the Russian Federation: since 04.12.2006 №200–FL

[2] Grigorev I V 2006 Reducing the negative impact on the soil of wheeled skid-ting tractors by justifying the modes of their movement and technological equipment (In Russia - Snizhenie otricateľ'noho vozdejstviya na pochvu kolesnych trelevochnykh traktorov obosnovaniem rezhimov ih dvizheniya i tehnologicheskogo oborudo-vaniya) (Saint-Petersburg: SpBFTA), p 236

[3] Grigorev I V and Zhukova A I 2004 Coordinate-volume tracing technique for the development of cutting areas by skidding News of higher educational institutions. Forest Magazine (In Russia - Koordinatno-obemnaya metodika trassirovaniya pri osvoenii lesosek trelevkoj) vol 4 pp 39-44

[4] Rudov S E, Voronova A M, Chemshikova J M, Tetereleva E V, Kruchinin I N, Don dokov Yu Zh, Khaldeeva M N, Burtseva I A, Danilov V V and Grigorev I V 2019 Theoretical approaches to logging trail network planning: increasing efficiency of forest machines and reducing their negative impact on soil and terrain Asian Journal of Water, Environment and Pollution vol 16 4 pp 61-75

[5] Grigorev I V, Kunieckaya O, Burgonutdinov A M, Burmistrova O N, Druzyanova V P, Dolmatov N P, Voronova A M and Kotov A 2020 Assessment the effect of skidding techniques on the ecological efficiency of the skidding tractor Diagnostyka vol 21 No 3 pp 67-75

[6] Grigorev I V, Kunitskaya O A, Prosuzhih A, Kruchinin I N, Shakirzyanov D I, Shvetsova V, Markov O B and Egipko S V 2020 Efficiency improvement of forest machinery exploitation Diagnostyka vol 21 No 2 pp 95-109

[7] Rudov S E, Grigorev I V, Kunitskaya O A, Druzyanova V P, Pekhutov A S, Ivanov A P, Ivanov A K, Okhlopkova M K, Pankov V Yu. and Borovikov R G 2019 Specific features of accounting of state of the massive of the frozen soil grounds under cyclic loads Bulgarian Journal of Agricultural Science vol 25 No S2 pp 191-205

[8] Manukovsky A Y, Grigorev I V, Ivanov V A, Gasparyan G D, Lapshina M L, Makarova Yu A, Chetverikova I V, Yakovlev K A, Afonichev D N and Kunitskaya O A 2018 Increasing the logging road efficiency by reducing the inten-sity of rutting: mathematical modeling Journal of Mechanical Engineering Research and Developments vol 41 No 2 pp 35-41
[9] Ivanov V A, Grigorev I V, Gasparyan G D, Manukovsky A Y, Zhuk A Yu, Kunitskaya O A and Grigoreva O I 2018 Environment-friendly logging in the context of water logged soil and knob-and-ridge terrain *Journal of Mechanical Engineering Research and Developments* vol 41 No 2 pp 22-27

[10] Glück R 2016 Covering Polygons with Rectangles (EuroCG, Lugano, Switzerland)

[11] Culberson J C and Reckhow R A 2002 Covering polygons is hard *29th Annual Symposium on Foundations of Computer Science* pp 601–611

[12] Rego G E and Voronov R V 2020 On Algorithms for the Minimum Link Disjoint Paths Problem 2020 *International Russian Automation Conference (RusAutoCon)* pp 525–529

[13] Rego G E 2020 On algorithms for the problem of finding disjoint paths with a minimum number of links (In Russia - Ob algoritmakh zadachi poiska neperesekayushchikhsya putey s minimal'nym chislom linkov) pp 185-190