

Development of Routing Methods for Cutting out Details ^{*}

Tatiana Makarovskikh and Anatoly Panyukov

South Ural State University, Chelyabinsk, Russia
Makarovskikh.T.A@susu.ru
paniukovav@susu.ru

Abstract. Laser cutting is one of the major cutting processes used to manufacture sheet metal products. Lots of researches on tool paths for cutting machines mainly deal with contour by contour cutting. While constructing a path one needs to determine the pierce point and the direction of contour passing. In this case only the length of idle passes may be optimized. To solve this problem generalized travelling salesman problem (GTSP) approach is used. Resource-efficient technologies for cutting sheet materials allow for the contours of cut-off details to be overlapped. It allows reducing the material waste and shortening the length of cuts. Common cuts are also the origin of one more set of precedence constraints. These constraints can be formalized as one general formal restriction called as Ordered Enclosing (OE) for plane graphs that are the homeomorphic images of the cutting plan. In this report we consider the common case of a cutting problem when combination of contours is allowed. We review the polynomial algorithms for all the possible restrictions: (1) part cut off a sheet does not require further cuts (constructing of OE-route); (2) there are no intersections of cuts (constructing of NOE-route); (3) there are some restrictions on placement of pierce points (constructing of PPOE-cover).

Keywords: Laser cutting · Cutting path · Tool path · Algorithm

1 Introduction

Laser cutting is one of the major cutting processes used to manufacture sheet metal products. A typical production process consists of designing parts, nesting the parts on metal sheets, and cutting the parts from the sheets [2]. After the assignment parts to a metal sheet computer aided manufacturing (CAM) software executes the actual nesting and generates cutting plan. In sheet metal laser

^{*} The work was supported by Act 211 Government of the Russian Federation, contract No. 02.A03.21.0011.

Copyright © by the paper's authors. Copying permitted for private and academic purposes. In: S. Belim et al. (eds.): OPTA-SCL 2018, Omsk, Russia, published at <http://ceur-ws.org>

cutting, a typical cutting process can take between several minutes to several hours, depending on the number of parts on the plate, the material type, the machine, the process parameters, and the plate thickness.

Lots of researches [2]–[1] on tool paths for cutting machines mainly deal with contour by contour cutting. In this case a part consists of an outer contour and possibly set of inner contours. Each contour itself is a cycle consisting of a finite set of lines and arcs. While constructing a path one needs to determine the pierce point and the direction of contour passing. In this case only the length of idle passes may be optimized. To solve this problem GTSP approach is used. In GTSP approach, the laser head can initiate a contour at a pre-defined set of points, but once a contour is initiated, it needs to be cut completely before moving on to another contour.

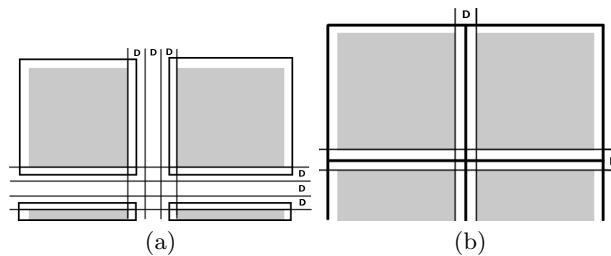


Fig. 1. The distance between details (a) without overlapping and (b) with it

Resource-efficient technologies for cutting sheet materials allow for the contours of cut-off details to be overlapped. It allows reducing the material waste and shortening the length of cuts (see fig.1). Nevertheless, some other problems arise.

Besides minimizing the above costs, the tool path problem is subject to precedence constraints coming from the fact that once a contour is completely cut, it detaches from the rest of the plate. The detached area can possibly shift position and hence becomes inaccessible for further cuts [2], [7]. However, there are a few contours held in place by clamps, they are not subject to these precedence constraints. In this paper these few contours are not considered.

These precedence constraints come from simple inner-outer contour relations, that can come from holes in parts, parts nested in holes, and parts nested in islands [2]. Islands be waste areas that have become completely enclosed by other parts because of common cut nesting. Common cuts are the result of efficient nesting algorithms that place parts so close to one another that they share elements. This has the benefit that only one cut must be made instead of two cuts.

Common cuts are also the origin of one more set of precedence constraints [2]. Each common cut is enclosed by a contour composed of the common cuts

two contours. The constraint is that common cut needs to be cut before its composite contour is completely cut.

These two constraints can be formalized as one general formal restriction called as Ordered Enclosing (OE) formulated in [7], [9], [10] for plane graphs that are the homeomorphic images of the cutting plan.

One more constraint is absence of intersections between cutter trajectories (touches are allowed). The reason is that after intersecting the trajectory cutter goes through a gap formed what may cause the loss of details quality. To formalize this restriction the definition of AOE-route is presented (the move continues by a neighbouring contour) and NOE-route (trajectories do not intersect but touches are allowed, and the move is not necessary to continue by a neighbouring contour). Routes belonging to classes AOE and NOE satisfy the condition of ordered enclosing.

Technological constraints may arise because of nesting. In this case some pierce points should be fixed. Hence, the recognition problem for opportunity of cutting (a route existence problem) for a given cutting plan arises.

For the tool path problem, elements are defined by their start and end nodes and can be cut in both directions. Thus, one does not need any information on the detail shape to define the sequence of detail cutting. This is true if one does not have to take thermal effects into account. Therefore, all curves without self-intersections and contiguities on a plane representing the shape of details are interpreted as graph edges, and all points of intersection and contiguity are graph vertices. One needs introduce additional functions to the set of vertices, faces, and edges of the graph received to analyse the satisfaction of technological restrictions [7].

Let us designate some technological definitions and make an adequate correspondence from graph theory terms.

Cut move (graph edge) be a movement where the laser head is actually cutting. The cutting time can be independent of the chosen tool path.

Air move (additional graph edge) be a movement where the laser head is not cutting. The time required for an air move is considerably less than the time required for the cut move.

Piercing (graph vertex) – every time the laser needs to start cutting in a new section of the sheet some piercing needs to be made. Path (edge-disjoint OE-cover by the ordered set of chains of a graph being the homeomorphic image of a cutting plan) be the instrument trajectory corresponding the shape and nesting of parts on the sheet.

The construction of a path to cut off each component is reduced to the construction of an ordered Eulerian covering for the given component by OE-chains [9], [10]. The order of routing within the components is defined by the solution to a generalized salesman problem on the digraph of allowed transitions between components with precedence constraints according to the nesting of one component to the contours of others. The restriction of components being Eulerian allows for increased quantity through overlapping fragments of contour parts and, consequently, decreases the number of components. This problem is

not easier or smaller, but, it is more difficult since these sub tool paths still need to be determined in conjunction with the idle passes between the components. These sub tool paths also contain idle passes and these sub tool paths are also determined by the higher-level path between components.

So, contour overlapping allows reductions in the material waste, the length of cutting, and the length of idle paths, due detail border combination.

In this paper the common case of a cutting problem when combination of contours is allowed [7]. We review the algorithms for all the considered conditions: part cut off a sheet does not require further cuts (constructing of OE-route); there are no intersections of cuts (constructing of NOE-route).

2 OE-Routes for Plane Graphs

To solve the stated problem cutting plan should be represented as a plane graph [7]. Let plane S be a model of a metal sheet, plane graph G be a model of cutting plan. Let $E(G)$ be a set of edges of graph G which are plane Jordan curves with pairwise disjoint interiors homeomorphic to open segments. The set of vertices $V(G)$ is represented by the set of bounding points of these curves.

For any $J \subseteq G$ (part of a cutter trajectory) let the theoretical-set union of its inner faces be designated as $\text{Int}(G)$ (the union of all its components $S \setminus J$ without outer face). Then $\text{Int}(G)$ may be interpreted as a part cut off a sheet. Let an initial part of a route in graph G be considered as a part of a graph containing all vertices and edges belonging to this part of a route. This allows formalizing the claim to a cutter as a condition of absence of initial route part inner faces of graph G intersection with unpassed graph G edges [11]. Such type of routes is called as OE-routes [9].

Definition 1. [11] Let chain $C = v_1e_1v_2e_2 \dots v_k$, $0 < k < |E(G)|$ so that $\text{Int}(v_1e_1v_2e_2 \dots e_l) \cap E(G) = \emptyset$, $1 \leq l \leq k$ be called ordered enclosing (or OE) chain.

Definition 2. [9] Let the ordered sequence of edge-disjoint OE-chains

$$C^0 = v^0 e_1^0 v_1^0 e_2^0 \dots e_{k_0}^0 v_{k_0}^0, \tag{1}$$

$$C^1 = v^1 e_1^1 v_1^1 e_2^1 \dots e_{k_1}^1 v_{k_1}^1, \dots, \tag{2}$$

$$C^{n-1} = v^{n-1} e_1^{n-1} v_1^{n-1} e_2^{n-1} \dots e_{k_{n-1}}^{n-1} v_{k_{n-1}}^{n-1}, \tag{3}$$

covering graph G and such that

$$(\forall m : m < n), \quad \left(\bigcup_{l=0}^{m-1} \text{Int}(C^l) \right) \cap \left(\bigcup_{l=m}^{n-1} C^l \right) = \emptyset,$$

be called cover with ordered enclosing (OE-cover).

Constructing of OE-route for graph G solves the stated above tool path problem. Routes with a minimum number of chains have the major interest since the transition from one chain to another corresponds to the idle cutter pass.

Definition 3. [7] Let minimal cardinality ordered sequence of edge-disjoint OE-chains for plane graph G be called Eulerian cover with ordered enclosing (Eulerian OE-cover).

One needs to define the following functions for each edge $E \in E(G)$ to represent the image of cutting plan as a plane graph $G = (V, E)$ [12]:

- $v_k(e)$, $k = 1, 2$ be vertices incident to edge e ;
- $f_k(e)$ be a face placed on the right-hand side when one is moving over edge e from vertex $v_k(e)$ to vertex $v_{3-k}(e)$, $k = 1, 2$;
- $l_k(e)$ be the edge incident to face $f_{3-k}(e)$ and $v_k(e)$, $k = 1, 2$;
- $r_k(e)$ be the edge incident to face $f_k(e)$ and $v_k(e)$, $k = 1, 2$.

As functions $v_k(e)$, $f_k(e)$, $l_k(e)$, $r_k(e)$, $k = 1, 2$, constructed on graph $G = (V, E)$ edges define incident vertices for each edge, incident faces and adjacent edges the following statement holds.

Theorem 1. [7] Functions $v_k(e)$, $f_k(e)$, $l_k(e)$, $r_k(e)$, $k = 1, 2$ constructed on graph $G = (V, E)$ edges define plane graph $G = (V, E)$ up to homeomorphism.

Further we consider that all considered plane graphs are represented by these functions. The space complexity of such a representation is $O(|E| \cdot \log_2 |V|)$. There is no problem to get these functions. In fact, they are defined on the stage of interpreting the cutting plan in terms of graph G . This is minimal information need to representation of any plane graph up to homeomorphism. Using the known coordinates of graph $G = (V, E)$ vertices images and nesting the fragments of the cutting plan (the images of graph $G = (V, E)$ edges) any route in graph may be interpreted as a tool trajectory.

Let us introduce the following definition of rank for graph G edges to formalize the technological restrictions on the order of cutting.

Definition 4. [7] Let rank of edge $e \in E(G)$ be a value of function $rank(e) : E(G) \rightarrow \mathbb{N}$ defined recursively:

- let $E_1 = \{e \in E : e \subset f_0\}$ be a set of edges bounding outer face f_0 of graph $G = (V, E)$ then $(\forall e \in E_1) (rank(e) = 1)$;
- let

$$E_k(G) = \{e \in E(G) \setminus \{\cup_{l=0}^k E_l\}\}$$

be a set of rank 1 edges for graph

$$G_k(V, E \setminus (\cup_{l=0}^k E_l))$$

then $(\forall e \in E_k) (rank(e) = k)$.

The rank of an edge defines its remoteness from the outer face and defines the minimal number of faces to be crossed to get from outer face f_0 to this edge. Algorithms for constructing OE-chains and OE-routes are presented in table 1.

Table 1. Algorithms for OE-routes constructing

Route type	Comp. complexity
Eulerian OE-cycle (alg. Recursive_OE) [11]	$O(V ^2)$
Eulerian OE-cycle (alg. OE-Cycle) [9], [10]	$O(E \cdot \log_2 E)$
OE-Postman Route (alg. CPP_OE) [12]	$O(E \cdot V)$
OE-Cover [7]	$O(E \cdot \log_2 E)$
Optimal OE-Cover [7]	$O(V ^2)$
OE-Cover for Disconnected graph [7] (alg. MultiComponent)	$O(E \cdot \log_2 E)$
OE-Cover for Disconnected graph [7] (alg. DoubleBridging)	$O(E \cdot \log_2 E)$

3 NOE-Covers for Plane Graphs

3.1 Definitions and Statements

Further we use some definitions from papers [9], [3], [13], [4]. Let us introduce them for completeness of presentation.

Definition 5. [13] Let graph of allowed transitions $T_G(v)$ of vertex $v \in V(G)$ be a graph which vertices are edges incident to vertex v , i.e. $V(T_G(v)) = E_G(v)$, and set of edges is represented by the allowed transitions between edges.

Definition 6. [13] Let system of allowed transitions T_G be the set $\{T_G(v) \mid v \in V(G)\}$ where $T_G(v)$ is the graph of transitions for vertex v .

Definition 7. [13] Let path $P = v_0, e_1, v_1, \dots, e_k, v_k$ in graph G be T_G -compatible if $\{e_i, e_{i+1} \in E(T_G(v_i))\}$ for each i ($1 \leq i \leq k-1$).

Definition 8. [13] [4] Let a cyclic order $O^\pm(v)$ is given for each vertex $v \in V(G)$ for chain $T = v_0, k_1, v_1, \dots, k_n, v_n, v_n = v_0$. This cyclic order defines the system of transitions $A_G \subset O^\pm(v)$. If $\forall v \in V(G) A_G(v) = O^\pm(v)$ the system of transitions $A_G(v)$ be called the full system of transitions.

Definition 9. [3] Let Eulerian chain T be called A -trail if it is an A_G -compatible chain. Thus, consequent edges from chain T (incident to vertex v) are neighbours in cyclic order $O^\pm(v)$.

Definition 10. [4] Let a chain be an AOE-chain if it is OE-chain and A -trail simultaneously.

Let us introduce the following theorems proved in [4], [5].

Theorem 2. [4], [5] If there is an A -trail for a plane graph G , then there is also AOE-chain.

Theorem 3. [4], [5] There exists AOE-chain for any plane connected 4-regular graph G .

Algorithm AOE-TRAIL [5] allows to find an AOE-chain for plane connected 4-regular graph any partial graph of rank k of which has no cut-vertices.

Definition 11. [5] *Partial graph G_k of G where $E(G_k) = \{e \in E(G) : \text{rank}(e) \geq k\}$ is called as partial graph of rank k .*

To define the cut-vertices the following properties of 4-regular graphs are used.

Statement 1. *A vertex incident to four edges incident to outer face is a cut-vertex.*

Statement 2. *The outer face of partial graph G_k is a union of all faces of rank k in graph G .*

Definition 12. [7] *Let the rank of face $f \in F(G)$ be the value of function*

$$\text{rank} : F(G) \rightarrow \mathbb{Z}^{\geq 0} : \text{rank}(f) = \begin{cases} 0, & \text{if } f = f_0, \\ \min_{e \in E(f)} \text{rank}(e), & \text{otherwise,} \end{cases}$$

where $E(f)$ be a set of edges incident to a face $f \in F$.

So, if in advance all the cut-vertices of partial graphs G_k are 'correctly' splitted then as a result we have a graph any partial graph G_k of which has no cut-vertices.

The sequence of splitting has no value, as soon as it is the local operation. The 'correct' splitting means that we move from one arc of a cyclic order to another and arcs belong to different pairs of faces (see fig. 2(a)). The result of splitting is presented in figure 2(b). This procedure may be realized by **Cut-Point-Splitting** algorithm [5] which considers vertices one by one and splits the cut-vertices of any rank.

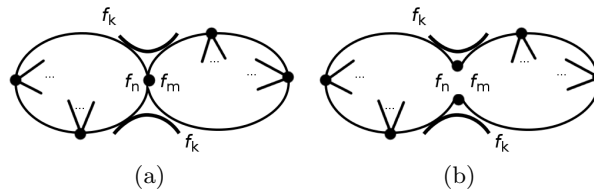


Fig. 2. (a) The correct transitions system for splitting a cut-vertex of partial graph G_k . (b) The result of splitting according to transitions system of the cut-vertex of partial graph G_k .

From all the above, the effectiveness of the **Cut-Point-Splitting** algorithm used to perform the splitting operation of cut-vertices of all ranks in a plane connected 4-regular graph follows [5].

Theorem 4. *Algorithm **Cut-Point-Splitting** needs time $O(|V(G)|)$ to identify and split all the cut-vertices of plane connected 4-regular graph $G = (V, E)$.*

Theorem 5. *Algorithm AOE-TRAIL allow to construct AOE-chain for a plane connected 4-regular graph G by time $O(|E(G)| \cdot \log_2|V(G)|)$.*

3.2 Class of NOE-chains and Algorithm for their Constructing

Class of AOE-chains describes all the trajectories of the cutting tool by the adjacent contour but does not cover completely all possible routes of the cutting tool with no intersections of cuts. Here the problem of non-intersecting OE-chain (see definition 14) arises.

Definition 13. [6] *Let Eulerian cycle of plane graph G be called non-intersecting if it is homeomorphic to a cyclic graph \tilde{G} obtained from graph G by $|E(G)|$ splittings of each vertex.*

Definition 14. *We say that chain belongs to class NOE-chains if it is OE-chain and non-intersecting chain simultaneously.*

Definition 15. *Let the transition system corresponding to non-intersecting chain be called a system of non-intersecting transitions.*

The proof of the fact of existence of such a starting vertex and finishing edge incident to outer face for a system of transitions corresponding to non-intersecting Eulerian cycle that allow to get the OE-cycle is like the proof of theorem 2 and gives the algorithm of such a chain constructing.

To get a non-intersecting Eulerian OE-chain (or a cycle) for a plane Eulerian graph without fixed transitions system (later this chain is called NOE-chain) one may act the following way.

Let's define a boolean function

$$\text{Checked}(v) = \begin{cases} \mathbf{true}, & \text{if the vertex is considered;} \\ \mathbf{false}, & \text{otherwise.} \end{cases}$$

on the set of graph G vertices.

Let all the vertices be unchecked on the stage `Initiate()`. Function `Non-intersecting(G)` (see alg. 1) splits all vertices $v \in V(G)$, $\deg(v) \geq 2n$, $n \geq 3$ of graph G to k pseudo-vertices of degree 4 and enters k additional pseudo-edges incident to these pseudo-vertices and forming a cycle.

Algorithm 1 Function Non-intersecting (G)

Require: plane Eulerian graph G ;
Ensure: plane connected 4-regular graph G^* ;

```

for all ( $e \in E(G)$ ) do                                ▷ Look through all the graph edges
     $k = 1$ ;                                              ▷ Consistently process function of index 1, later - 2
    while ( $k \leq 2$ ) do
        if NOT(Checked( $v_k(e)$ )) then                    ▷ If vertex is not processed
            Handle ( $e, v_k(e), k$ );                          ▷ Process the vertex
        end if
         $k + +$ ;
    end while
end forReturn  $G^*$ ;

```

To realize this transfiguration, one needs to look through all the functions $v_k(e)$, $k = 1, 2$ for all the edges and modify the graph encoding. Procedure **Handle** ($e, v_k(e), k$) (see alg. 2) processes each unchecked vertex of G .

Algorithm 2 Procedure Handle (e, v, k)

```

1:                                ▷ Passage 1: Defining of vertex  $v$  degree
2:  $e_{first} = e$ ;
3:  $d = 0$ ;                          ▷ Variable  $d$  saves the vertex degree
4: repeat
5:    $le = l_k(e)$ ;
6:   if ( $v_k(le) \neq v$ ) then
7:     REPLACE( $le$ );
8:   end if
9:    $e = le$ ;  $d = d + 1$ ;
10: until ( $e = e_{first}$ );
11:                                ▷ Passage 2: Splitting of vertices with degree greater than 4
12: if ( $d > 4$ ) then
13:    $e = e_{first}$ ;  $le = l_k(e)$ ;  $fl = \text{new EDGE}$ ;  $fle = fl$ ;  $e_{first} = e$ ;  $e_{next} = l_k(le)$ ;
14:   repeat
15:      $e = e_{next}$ ;  $le = l_k(e)$ ;  $fr = fl$ ;  $fl = \text{new EDGE}$ ;  $e_{next} = l_k(le)$ ;
16:     Pointers( $e, le, fr, fl$ );                                ▷ Put the edge pointers
17:   until ( $l_k(le) = e_{first}$ );
18:   Pointers( $e_{first}, l_k(e_{first}), fle, fe$ );                ▷ Put the edge pointers
19: end if

```

Processing of vertex $v_k(e)$ means its splitting according to figure 3(a, b).

If degree of checked vertex is equal to $2k$ then k pseudo-vertices entered by procedure **Handle** and k pseudo-edges incident to these vertices form a cycle. As a result of processing of all the graph G vertices we get a modified graph G^* which is a plane connected 4-regular graph. Hence, the following theorem holds.

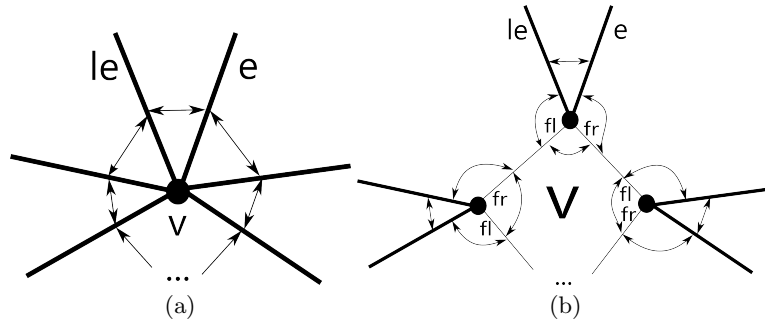


Fig. 3. (a) Initial pointers to the neighboring edges of splitted vertex. (b) Splitting of a vertex (bold lines are graph G edges, thin lines correspond to additional edges (pseudo-edges)) and pointers modification according to this splitting.

Theorem 6. *Function **Non-Intersecting** allows to reduce any plane connected Eulerian graph G to a plane connected 4-regular graph G^* by time $O(|E(G)| \cdot \log_2|V(G)|)$.*

Algorithm **AOE-trail** may be used for definition of AOE-chain T^* . If to delete all pseudo-edges and absorb all splitted vertices to v then we get a non-intersecting chain T for graph G . The obtained chain belongs to OE-class because the procedure of absorbing the vertices does not vanish the sequence of edges in the chain what excludes appearing of a cycle enclosing the unpassed edges. Computing complexity of this reduction algorithm is $O(|E(G)| \cdot \log_2|V(G)|)$. Hence the following theorem holds.

Theorem 7. *The computational complexity of constructing a NOE-route for plane graph (V, E) does not exceed $O(|E(G)| \cdot \log_2|V(G)|)$.*

4 PPOE-Covers and their Existence

One of the most common cutting technologies is a plasma cutting technology. Nowadays the most common method for cutting of details using plasma cutting is GTSP-technology (contour by contour cutting). The technology using combination of cuts is not widely used now. Nevertheless it allows save material and reduce the cut length. Plasma cutting technology puts some additional restrictions on the instrument path. One of the major restrictions is necessity of leaving some free space for placement of pierce points. Besides time need for cutting significantly affects on the time of cutting.

The problem of feasibility of cutting with plasma cutting technology for solving the cutting-packing problem arises due to these restrictions. The problem of minimization of pierce points number while constructing the cutter path also arises.

As soon as minimal number of pierce points for a cutting map represented by a plane connected graph is equal to $|V_{odd}|/2$ in this section we consider only

graphs coverable by $|V_{odd}|/2$ chains, i.e. bridgeless graphs with at least one vertex incident to outer face.

Let us consider the cutting plans in fig. 4. We admit that cuts combining technology was used for placement of rectangular parts. So, piercing can be realized from the vertices incident to outer face (in common we may use any face allowing piercing).

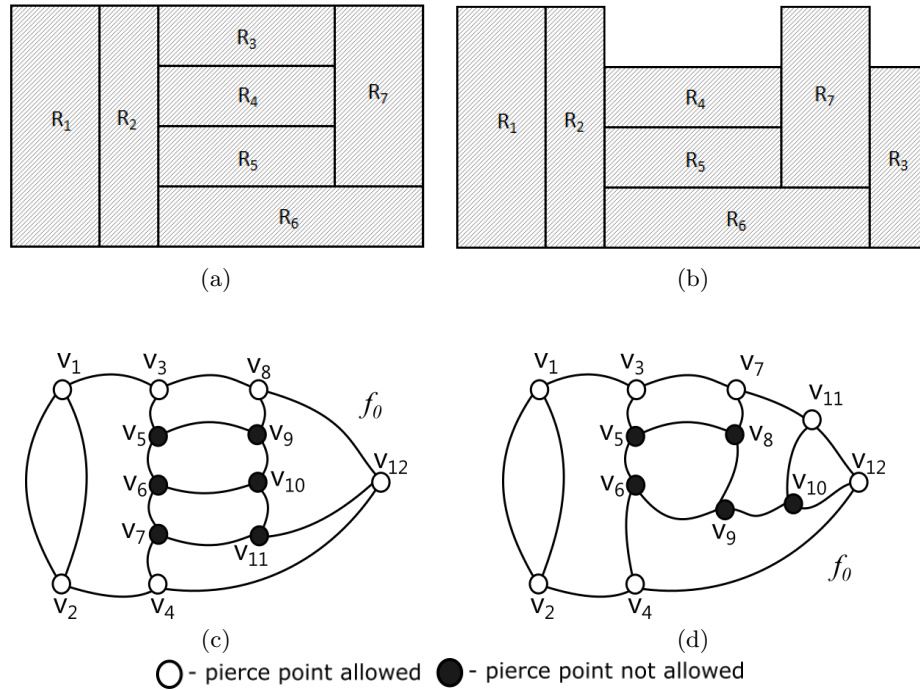


Fig. 4. The example of an unrealizable and feasible cutting for plasma cutting technology

Hence, realization of cutting by plasma automata is possible for packing in figure 4(b) and not possible for packing in figure 4(a). We need placement of additional pierce points for cutting the inner rectangles R_4 and R_5 in fig. 4(a). As for cutting plan in figure 4(b) it is possible to place pierce points near the outer contour.

Let us formalize this problem.

Let faces $F_{in}(G) \subset F(G)$ allow placement of pierce point. Then let us designate vertices of off degree incident to $F_{in}(G)$ as $V_{in}(G) \subset V(G)$. If a route in graph is OE -route and starting vertex $v_1 \in V_{in}(G)$ then this route may be

a base for constructing the cutting program for plasma cutting machine. This type of routes be called as *PPOE*-route [8].

Definition 16. Let chain $C = v_1e_1v_2e_2 \dots v_k$ be called *PPOE-chain* if it is an *OE*-chain and starting vertex $v_1 \in V_{in}(G)$.

Definition 17. Let *PPOE-cover* of graph G be the *OE*-cover of graph G consisting of *PPOE*-chains.

Definition 18. Minimal cardinality ordered sequence of edge-disjoint *PPOE*-chains in plane graph G is called **Eulerian PPOE-cover**.

Graphs in figure 4(c) and (d) are the images of cutting maps in figures 4(a) and (b). Vertices V_{in} are designated by white circles. These vertices allow placement of pierce point nearby. On the other hand placement of pierce points near the black-marked vertices is impossible. Thus, Eulerian *PPOE*-cover exists for graph in figure 4(d). For example, it can consist of the following chains: $C_1 = v_1v_3v_5v_6v_9v_8v_5$, $C_2 = v_3v_7v_8$, $C_3 = v_7v_{11}v_{10}v_9$, $C_4 = v_{11}v_{12}v_{10}$, $C_5 = v_{12}v_4v_6$, $C_6 = v_4v_2v_1v_2$. Such a cover does not exist for graph in figure 4(c).

The problem of definition the possibility of cutting by plasma instrument may be stated as a problem of checking the existence of Eulerian *PPOE*-cover for a given graph. According to the mentioned restrictions we may formulate the following necessary condition of *PPOE*-cover existence.

Statement 3. Let G be a plane graph with $2k$ odd degree vertices. If there exists Eulerian *PPOE*-cover then $|V_{in}(G)| \geq k$.

For example, the graph in figure 5(a) cannot be covered by *PPOE*-chains. It has eight odd degree vertices and can be covered by minimum four Eulerian chains, and only three of them can start from vertices marked as starting ones for *PPOE*-chain. As for graph in figure 5(b) there are four vertices that can be starting for *PPOE*-chain and the same number of finishing vertices. Nevertheless, there is no *PPOE*-cover for this graph.

Paths realizing *PPOE*-cover can be represented by a special way ordered set of *PPOE*-chains with additional idle paths (edges) between the end of the current chain and beginning of the next one. Such transitions form a matching on a bipartite oriented graph $D = (V_{in} \cup v_{out} \rightarrow V_{in}, E)$ where V_{in} is a set of odd degree vertices allowed to be the beginnings of trails (pierce points); V_{out} is a set of odd degree vertices allowed to be only the ends of constructed chains (leaving points).

Statement 4. It is necessary for existing *PPOE*-cover to a mixed graph $G \cup D$ to have a cycle all additional arcs of which belong to

$$\{(v, u) : v \in V_{out} \cup V_{in}, u \in V_{in}\}.$$

Proof. *PPOE*-cover is a partial case of *OE*-cover, hence, it is an oriented cycle consisting of graph G edges and edges of matching M on set of vertices

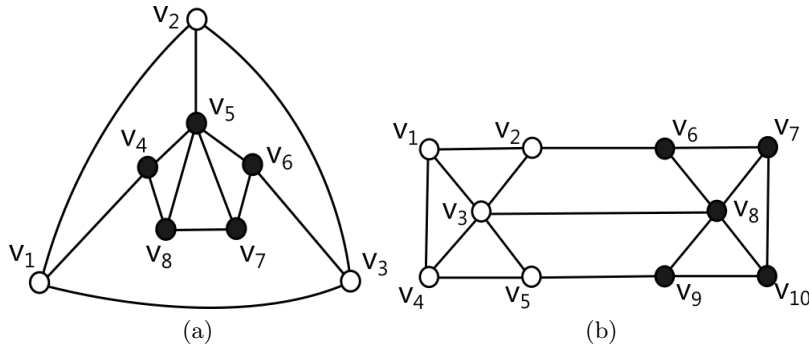


Fig. 5. Examples of graphs having no PPOE-cover

$V_{odd} \in G$ (vertices $V_{in} \cup V_{out}$). As soon as PPOE-cover consists of PPOE-chains then edges of matching M are to be passed in direction $V_{out} \cup V_{in} \rightarrow V_{in}$, hence, they correspond to arcs in D .

This yields that it is necessary for existing of PPOE-cover the existence of a cycle for mixed graph $G \cup D$ in which all additional edges are arcs from $V_{out} \cup V_{in}$ to V_{in} , as soon as PPOE-cover has this type of the cycle. **Statement now follows.**

Statement 5. *It is necessary for existence of PPOE-cover of plane connected graph G for cardinality of minimal $\{V_{in}, V_{out}\}$ -cut be not more than $|V_{out}|$.*

Proof. Let there exists PPOE-cover for graph G . Nevertheless, the cardinality of $\{V_{in}, V_{out}\}$ -cut is less than $|V_{out}|$. As soon as no one of PPOE-chains forming a cover cannot start in $u \in V_{out}$ then a cover consists of not less than $|V_{out}|$ ways from $v \in V_{in}$ to $u \in V_{out}$. Then some of these ways may be edge-disjoint what leads to a contradiction with definition of PPOE-cover. **Statement now follows.**

The graph in figure 5(b) can be a good example for statement above if we assume that we can begin PPOE-chain only from white vertex. The graph has ten odd degree vertices so we need minimum five vertices from where Eulerian chain can begin. Five of vertices allow it, but we cannot cover the graph by chains which are began only from these vertices. Not more then three chains, for example, $C_1 = v_5v_9v_8v_{10}v_9$, $C_2 = v_3v_8v_7v_6v_8$, $C_3 = v_4v_3v_2v_6$ that can be constructed so that chain begins from white vertex and ends in black one. The minimal cut between black and white vertices has three edges.

Constructing of PPOE-cover for graph G solves the routing problem for a cutter with restrictions on pierce points placement.

5 Conclusions

The technology of details' edge combination is actual resource-saving cutting technology. However, a few algorithms for its implementation exist. So the question of constructing these algorithms is currently important.

In this article we discussed the technology of details' edge combination from the point of view of cutting tool routing. We raised the question of the possibility of details cutting by plasma cutting machine and presented necessary conditions for it. But the issue of multiple using of one vertex for piercing or finishing a chain and the issue of using even degree vertices for piercing are not considered in this paper and present the aim of further researches.

References

1. Chentsov, A., Khachay, M., Khachay, D.: Linear time algorithm for precedence constrained asymmetric generalized traveling salesman problem. *IFAC-PapersOnLine* 49, 651–655 (2016)
2. Dewil, R., Vansteenwegen, P., Cattrysse, D.: Construction heuristics for generating tool paths for laser cutters. *International Journal of Production Research* 52(20), 5965–5984 (2014)
3. Fleischner, H.: Eulerian graphs and related topics. Part 1. *Ann. Discrete Mathematics* 50(2), 280 (1991)
4. Makarovskikh, T.: *A*-trails and their application to industrial process. In: 2nd International Conference on Industrial Engineering, Applications and Manufacturing (2016)
5. Makarovskikh, T., Panyukov, A.: *AOE*-trails constructing for a plane connected 4-regular graph. In: Supplementary Proceedings of the 9th International Conference on Discrete Optimization and Operations Research and Scientific School (DOOR 2016). Vladivostok, Russia, September 19 - 23, 2016. *CEUR Workshop Proceedings*. vol. 1623, pp. 62–71 (2016). <http://ceur-ws.org/Vol-1623/paperco11.pdf>
6. Makarovskikh, T., Panyukov, A.: The cutter trajectory avoiding intersections of cuts. *IFAC-PapersOnLine* 50(1), 2284–2289 (2017)
7. Makarovskikh, T., Panyukov, A., Savitskiy, E.: Mathematical models and routing algorithms for economical cutting tool paths. *International Journal of Production Research* 56(3), 1171–1188 (2018). <https://doi.org/10.1080/00207543.2017.1401746>
8. Makarovskikh, T., Savitskiy, E.: The *OE*-cover for a plane graph by chains with allowed starting vertices. In: *CEUR Workshop Proceedings*. vol. 2064, pp. 103–111 (2017). <http://ceur-ws.org/Vol-2064>
9. Panyukova, T.: Chain sequences with ordered enclosing. *Journal of Computer and System Sciences International* 46(1), 83–92 (2007)
10. Panyukova, T.: Eulerian cover with ordered enclosing for flat graphs. *Electronic Notes in Discrete Mathematics* 28, 17–24 (2007)
11. Panyukova, T., Panyukov, A.: Algorithms for construction of ordered enclosing traces in plane eulerian graphs. In: *The International Workshop on Computer Science and Information Technologies, Proceedings of Workshop*, vol. 1, pp. 134–138. Ufa State Technical University, Ufa State Technical University Publ. House, Ufa, Russian Federation (2003)

12. Panyukova, T.: Constructing of OE -postman path for a planar graph. Bulltein of South Ural State University. Series: Mathematical Modelling and Programming 7(4), 90–101 (2014)
13. Szeider, S.: Finding paths in graphs avoiding forbidden transitions. Discrete Applied Mathematics 126, 261–273 (2003)