A Novel Algorithm for Construction of the Shortest Path Between a Finite Set of Nonintersecting Contours on the Plane

Alexander Petunin[1,2], Efim Polishchuk[1], and Stanislav Ukolov[1](✉)

[1] Ural Federal University, Yekaterinburg, Russia
s.s.ukolov@urfu.ru
[2] Institute of Mathematics and Mechanics, UBr RAS, Yekaterinburg, Russia
https://urfu.ru/
https://www.imm.uran.ru/

Abstract. An optimization problem that arises during tool path routing for CNC sheet cutting machine is considered for the case when parts are bounded by line segments and circular arcs and pierce points lay on the bounds. Technique of continuous cutting is used, i.e. each contour is cut as a whole from any starting point. The task of tool path length minimization is reduced to the task of air move length minimization which is shown to be equivalent to finding the shortest broken line with vertices on non-nesting disjoint contours on the plane. The algorithm of building such a broken line for a fixed order of contour processing is devised and proved to deliver local minimum. Some sufficient conditions for this minimum to be global are discussed. A heuristic algorithm for finding the optimal contour cutting order is proposed based on Variable Neighborhood Search approach. Results of a computational experiment and a comparison with the exact solution of GTSP problem are presented.

Keywords: Tool path problem · Continuous cutting problem · Local search · Sufficient conditions of global extremum · Heuristic · Discrete optimization · Variable neighborhood search · GTSP

1 Introduction

A number of optimization problems arise during development of control programs for CNC sheet cutting machines. One of them is the task of minimizing the tool air move, which in some special cases can be reduced to the problem of finding the shortest polyline with vertices on flat contours. Contours are interpreted as the boundaries of flat parts. The location of the contours on the plane is determined during the solution of the "nesting" problem. Both tasks are generally NP-hard.

This work was supported by the Russian Foundation for Basic Research under Grant 20-08-00873.

N. Olenev et al. (Eds.): OPTIMA 2020, CCIS 1340, pp. 70–83, 2020.
https://doi.org/10.1007/978-3-030-65739-0_6

In its turn, the task of minimizing tool air move is a subtask of another optimization problem – the task of optimizing the tool path when cutting flat parts. Its exact solution cannot be obtained for problems that actually arise in production (for hundreds of parts/contours) in a reasonable time, therefore, various heuristics are typically applied to get solutions of acceptable quality. At the same time, the issues of developing algorithms that provide optimal solutions for some problem cases, as well as evaluating the quality of their solutions in comparison with the optimal solution, remain unresolved and are of significant scientific interest.

The general problem of optimizing the tool path when cutting 2D objects on CNC machines, which consists in minimizing cutting time and cost, includes a whole range of different optimization tasks. A classification of such problems can be found in [9,13,22], see Fig. 1.

- Continuous Cutting Problem (CCP): each closed contour (that bounds a part) is cut out entirely by one movement of the torch, but cutting can start from any point (and finishes at the same point).
- Generalized Traveling Salesman Problem (GTSP): cutting can start only at one of the predefined points on the contour, the contour must be cut entirely.
- Endpoint Cutting Problem (ECP): cutting can start only at one of the predefined points on the contour, and the contour can be cut in several approaches, in parts.
- Segment Continuous Cutting Problem (SCCP): the notion of a cutting segment is introduced, which is a generalization of a contour; it can be either a part of a contour or a combination of several contours or their parts. Each segment is cut out entirely, thus $CCP \subset SCCP$.
- Generalized Segment Continuous Cutting Problem (GSCCP): segment cutting (SCCP), but the selection of segments is not fixed in advance, but is subject to optimization
- Intermittent Cutting Problem (ICP): the most general cutting problem described in the literature, when contours can be cut in parts, in several approaches, and cutting can begin at any point in the contour.

Tool path optimization problems in practice often reduce to discrete optimization problems by discretizing the contours to be cut with a certain step ε, that is, they reduce to ECP [8,14,24] or its special case, GTSP [3,18,27,28]. CCP can also be reduced to GTSP. In this case, however, the total error in the air move length reaches $N \cdot \varepsilon$, where N is the number of contours. To guarantee the accuracy of the result of δ, it is necessary to choose a small $\varepsilon \approx \delta/N$, so the total number of points on the contour grows (as $O(N)$) and the exhaustive search becomes exponential. Nevertheless, such problems can be successfully solved, for example, by the dynamic programming (DP) method, for small $N \approx 30$ even precisely (see, in particular [6]).

Tool path routing without using discretization (CCP) is further considered in this paper. The publications on this subject are rare. [1,26] can be noted, where heuristic algorithms are proposed.

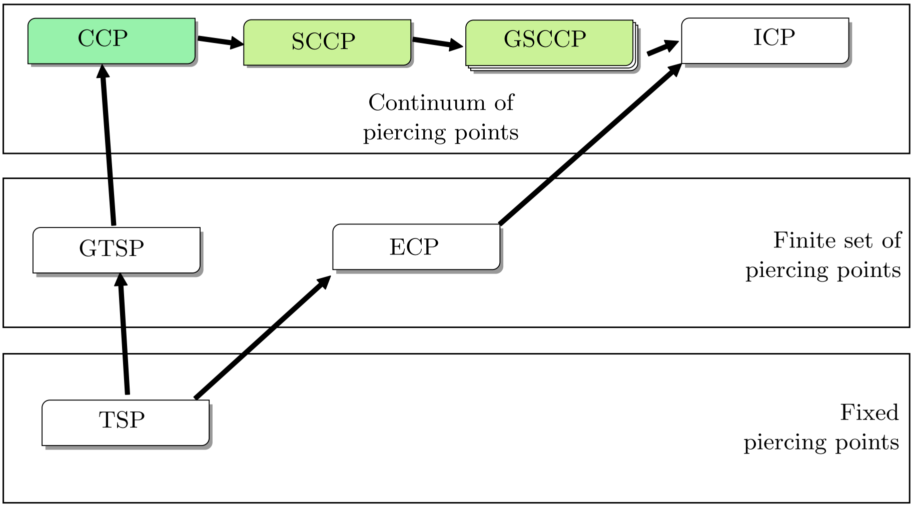


Fig. 1. Classification of Cutting Problems

1.1 Technological Constraints

The need to execute the tool path on a CNC sheet cutting machine imposes a number of technological limitations on it.

The so-called “precedence constraint” is by far most popular in the literature. It is caused by the fact that after cutting a closed contour, its interior is usually not held by anything and can freely shift, rotate and even fall. For this reason, the internal contours of parts must be cut before the external contours containing them, and parts located in the holes of large parts even earlier.

Finally, most cutting technologies require that the cutting not be carried out strictly along the contour, but with some indentation. This shift can be performed both during the solution of the routing problem, and after – at the stage of generating the control program for the CNC cutting machine or even by the machine itself during the cutting process. In addition, the pierce point (tool switch-on point) should generally be located even further from the contour to avoid part damage. However, this work completely ignores this requirement. Thus, it is further assumed that the tool moves exactly along the contour of the part and the pierce point is located directly on the contour (as well as the switch-off point of the tool).

2 Continuous Cutting Problem

Consider the Euclidean plane $\mathbb{R}^2$ and its region B bounded by a closed contour (rectangle in most cases), which is a model of the sheet material to be cut. Let N pairwise disjoint flat contours $\{C_1, C_2, ..., C_N\}$ be given inside B, bounding

n parts $\{A_1, A_2, ..., A_n\}$. A part can be limited by either one contour or several (external and internal holes), so that in general $n \leq N$.

The contours C_i can have an arbitrary shape, but we will only consider the case when they consist of (a finite number of) segments of lines and arcs of circles, which is determined by the existing technological equipment. In case when the contours consist only of line segments, the continuous cutting problem is reduced to one of the variants of the Touring Polygon Problem (TPP), see [10].

Further, two points are set in region B (usually at its boundary), we denote them as M_0, M_{N+1} (almost always $M_0 = M_{N+1}$), which represent the beginning and end of the cutting route.

Continuous Cutting Problem is to find:

1. N pierce points $M_i \in C_i, i \in \overline{1, N}$
2. Contour C_i traversal order, i.e. permutations of N elements $I = (i_1, i_2, ..., i_N)$

The result of solving the problem will be the route $\{M_0, M_{i_1}, M_{i_2},\ \ldots M_{i_N}, M_{N+1}\}$. The objective function in this case is greatly simplified in comparison with the general cutting problem and is reduced to minimizing the air move length.

$$\mathcal{L} = \sum_{j=0}^{N} |M_{i_j} M_{i_{j+1}}| \tag{1}$$

$$\mathcal{L} \to \min$$

Where, for sake of simplicity, we introduce the notation $M_{i_0} = M_0$, $M_{i_{N+1}} = M_{N+1}$.

In addition, we will solve the optimization problem with an additional constraint, the so-called "precedence constraint". Although the contours C_i do not intersect, they can be nested into each other, i.e., $\tilde{C_a} \subset \tilde{C_b}$, where $\tilde{C_a}$ denotes a 2-dimensional figure bounded by the contour C_a (in the more familiar notation $C_a = \partial\tilde{C_a}$). In the general tool path routing problem, this can be caused by two different circumstances (holes in parts and placement of smaller parts in holes larger to save material), but in this case these options are processed the same way.

If one contour is located inside another, then the nested contour must be cut out (visited) earlier than the outer one: $\tilde{C_a} \subset \tilde{C_b} \Rightarrow i_a < i_b$ in the permutation $I = (i_1, i_2, ..., i_N)$. Thus, not all permutation of the contours are feasible.

3 CCP-Relax Algorithm to Solve Continuous Cutting Problem

The proposed solution algorithm consists of several stages, easily associated with the nature of the problem being solved:

1. **Removal of external contours.** To automatically comply with the precedence constraint, we start by removing all contours containing nested ones. This generally leads to a reduction (significant in some cases) of the size of the problem (from N to some N'), and thus reduces the calculation time in the second and especially the third stage.
2. **Continuous optimization**. Assuming the order of contours processing $I = (i_1, i_2, ..., i_N)$ fixed we look for the coordinates of the pierce points $M_i \in C_i$, minimizing the total air move length (1).
   For every pierce point M_i we find it's optimal position, while others remain motionless. This relaxation is repeated a few times until converged. In practice, it happens very fast in $O(N)$ time and is therefore used as a subroutine in the next step.
3. **Discrete optimization**. We use Variable Neighborhood Search (VNS, see [12]) to find contours processing order $I = (i_1, i_2, ..., i_N)$.
   This step in fact solves famous *Travelling Salesman Problem* with special distance function, calculated at the previous step:
   $$\mathcal{L}(I') = \min_{M_1, M_2 \ldots M_N} \mathcal{L}(M_1, M_2 \ldots M_N | I')$$
   Note, that other heuristics for discrete optimization may be used at this step as well. For instance, one can use modern solvers to first solve GTSP problem, associated with CCP, and then apply continuous relaxation (previous step) to convert solution of GTSP to that of CCP. This idea deserves further investigation.
4. **Recovery of removed contours**. Having got the tool path that visits "inner" contours (remained after first step), we find piercing points for other contours by simple intersecting them with the tool path. Of multiple points we select one (for each contour) so as to meet precedence constraint.
   This is straightforward step of linear time complexity.

For detailed explanation of the CCP-Relax algorithm steps refer to [20].

3.1 Optimality of Continuous Optimization Problem Solution

From a practical point of view, the described algorithm turns out to be quite workable – it generates high-quality tool path routes in an acceptable time, but this is an empirical result. The theoretical justification of the properties of the resulting routes is interesting. The greatest difficulty is, of course, the third step of the algorithm – discrete optimization, both from a theoretical and a practical point of view. This work focuses the second step of the algorithm – continuous optimization.

Remark 1. Figure 2 shows an example where a trajectory that is not improved by shifts of vertices individually may not deliver a global minimum.

We were able to formulate some statements regarding the quality of continuous optimization solutions at Step 2 of CCP-Relax algorithm. We present them here without proof, which will be published in a separate paper.

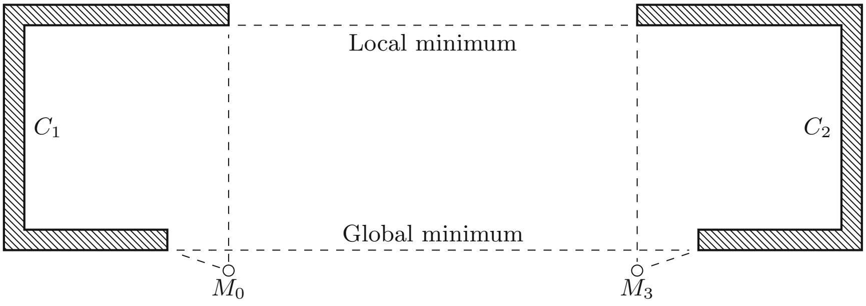


Fig. 2. Two tool paths delivering local minimum

We consider the case of fixed order of contours processing $I = (i_1, i_2, ..., i_N)$. and apply Step 2 of CCP-Relax algorithm to get broken line L_*, visiting all the contours C_i in the said order.

Proposition 1. *If we move several adjacent vertices of the broken line L_* so that they remain on the same segments of the contours, then the length of the resulting broken line will not decrease.*

This statement means that the algorithm always delivers a local minimum, however not yet global, as for example in Fig. 2.

To guarantee the latter, the following sufficient condition may be required:

Condition 1. *Let **one** of the following requirements be satisfied for every piercing point M_i:*

1. *Segment $M_{i-1}M_{i+1}$ intersects the contour C_i, i.e. $M_i \in M_{i-1}M_{i+1}$*
2. *The tangent at M_i to the ellipse with foci M_{i-1} and M_{i+1} and passing through M_i separates the ellipse and the contour C_i.*

Proposition 2. *Let Condition 1 is satisfied for (every vertex of) L_*.*

If we move several adjacent vertices of the broken line L_* so that they remain on the contours, then the length of the resulting broken line does not decrease, that is, the broken line L_* delivers a global minimum.

Remark 2. Suppose that besides the trajectory L_*, there is another trajectory delivering a global minimum. Then it follows from the last statement that they coincide as lines, that is, the difference can only be at the points of intersection with the contours.

Condition 1 is easily verified programmatically, but it can be simplified so that in most practical cases to be checked simply visually.

Condition 2. *When segment $M_{i-1}M_{i+1}$ doesn't intersects the contour C_i but*

1. *If the vertex M_i is the internal point of the linear segment of the contour and the entire contour C_i is on one side of the that segment line (which is the tangent from Condition 1; otherwise there must be a better $M_i' \in C_i$).*
2. *If the vertex M_i is terminal (belongs to two linear segments of the contour; is also vertex of C_i), and the entire contour is inside the corner with the rays from the point M_i along these segments.*
3. *If the region $\tilde{C}_i$ bounded by the contour C_i is convex.*

4 New Approach to Intermittent Cutting Problem

Intermittent Cutting Problem is the most complex and general of all varieties of cutting problems. It can be approached to both from theoretical positions as well as by using some practical techniques.

In addition to standard cutting technique (which in fact leads to Continuous Cutting Problem), some others are often used, for instance, "multi-segment" and "multi-contour" cutting. The former cuts single contour of a part in several passes, using several piercing points. The latter cuts a few contours at once, as seen at Fig. 3.

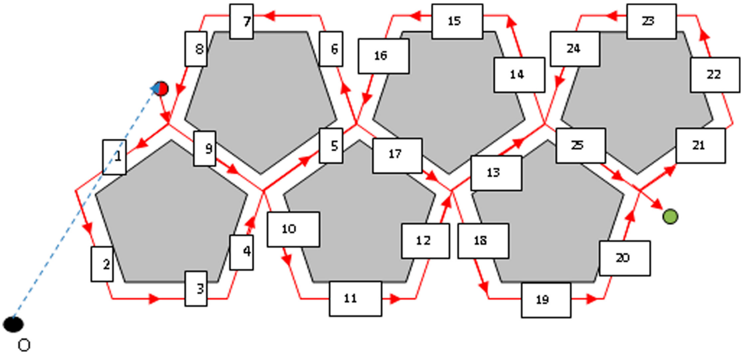


Fig. 3. Example of complex cutting segment for six parts/contours

In order to apply those technique we introduce a notion (see [21]):

Cutting Segment $S = \overrightarrow{MM^*}$ is a tool trajectory from piercing point M up to point of switching tool off M^*.

Cutting segment is used to contain single contour, but this is not the case any more. It also can be a part of contour (for multi-segment cutting) as well as several contours at once (i.e. multi-contour cutting).

In fact, multi-contour cutting example at Fig. 3 can also represent a single cutting segment in some bigger cutting problem instance.

Since the cutting direction is defined for the cutting segment, we need a more general concept:

Basic Segment B^S is a part of cutting segment $S = \overrightarrow{MM^*}$ without lead-in and lead-out trajectory (the very beginning and ending parts of segment, where tool approaches contour of a part and leaves it). Basic segment has no direction and contains only geometry of contours to cut.

Using the concept of basic segment, we can now formulate a generalization of CCP:

Segment Continuous Cutting Problem ($SCCP$) is a cutting problem with fixed set (as well as number of) basic cutting segments: $SCCP = \left\{B^{S_i}\right\}$.

CCP-Relax algorithm described above can be applied to solve SCCP problem in the same way as for CCP problem for which it was originally designed.

And now, note that for predefined nesting (i.e. fixed positioning of parts' contours on the plain), the whole *ensemble* of basic segments can be generated by combining and dividing contours into different segments. See, for instance, Fig. 4, where multi-contour segments are filled with black color. This leads us to even more general:

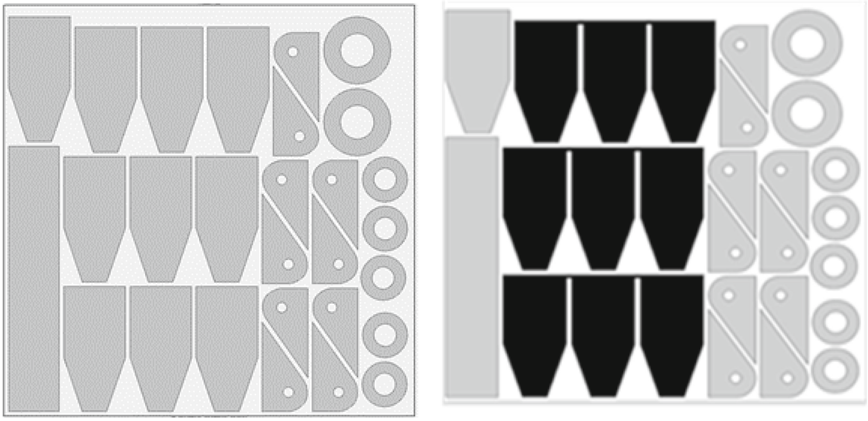

(a) Standard cutting, 45 segments (b) Multi-contour cutting, 39 segments

Fig. 4. *Ensemble* of Segment Cutting Problems

Generalized Segment Continuous Cutting Problem ($GSCCP$) is that *ensemble* of several $SCCP$ problems for the same nesting: $GSCCP = \{SCCP_i\}$.

By introducing the class of $GSCCP$, we have significantly expanded the existing classification of tool path problem for CNC sheet cutting machines. Actually $SCCP$ and $GSCCP$ are ICP subclasses containing all tasks with finite sets of basic cutting segments, i.e. $CCP \subset SCCP \subset GSCCP \subset ICP$.

General Scheme for GSCCP Solving

Assuming an ensemble $\{SCCP_i\}$ of base segment sets $SCCP_i = \left\{B^{S_j}\right\}$, $i \in \overline{1, T}$, $j \in \overline{1, K_i}$ to be known, the following scheme for GSCCP solving is presented:

- Each task $SCCP_i$ is solved independently with one of existing algorithms, for instance:
  1. *CCP-Relax*, heuristic described above in Sect. 3.
  2. *DP-GTSP*, exact algorithm based on Dymaic programming for the case of relatively small problem dimensions, see [6]
  3. *Greedy-GTSP*, iterative greedy heuristic algorithm, see [19]

  For discrete algorithm to use, cutting segments can be pre-sampled as shown at Fig. 5.
- The best solution is selected according to objective function (1).

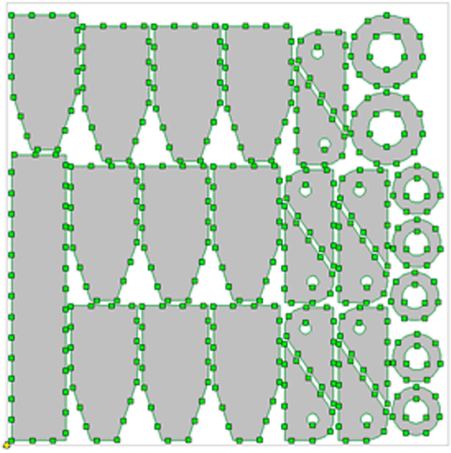
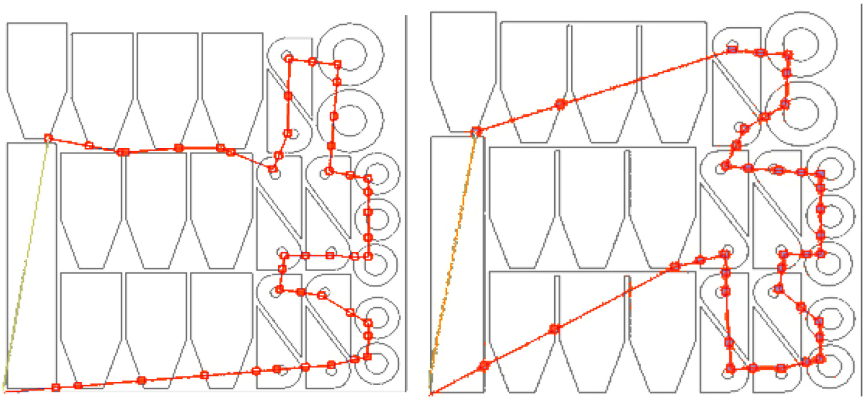


Fig. 5. Corresponding GTSP problem for (S)CCP problem of Fig. 4, 425 points

For example, Fig. 6 shows two solutions of $SCCP$ problems from Fig. 4 given by *CCP-Relax* algorithm. It is easy to see that the two routes are indeed different. Furthermore, the difference can be even more significant in a practical sense due to different numbers of piercing points, since that operation is rather expensive both in terms of time and cost.

5 Numerical Experiments

The quality assessment of the solutions of the described algorithm was carried out on several cutting plans containing real parts. As a comparison base, we used DP algorithm (see [6]) for solving the GTSP problem, which gives an exact solution for small number of contours and special version of GNLS heuristic [25].



(a) Standard cutting

(b) Multi-contour cutting

Fig. 6. Solution of GSCCP Problem at Fig. 4

Figure 7 shows the exact solution, possible positions of the pierce points are visible. Figure 8 shows the solution to the CCP problem for the same cutting plan.

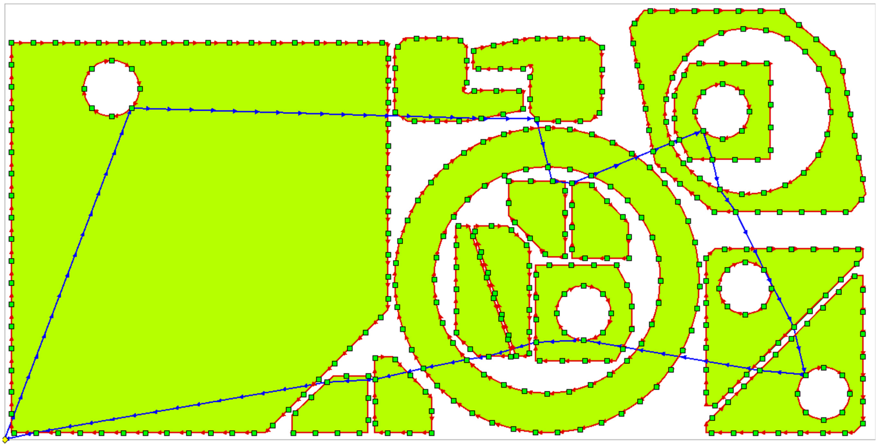


Fig. 7. Exact solution of GTSP, Job #464

It can be seen that both algorithms generated almost identical routes. The main difference is caused by the discretization process to obtain the GTSP task. Because of this, the segments of the route that are straight in the CCP solution turn out to be slightly broken in the GTSP solution, hence total air move length is slightly larger. Numerically, this is shown in Table 1 for several cutting plans.

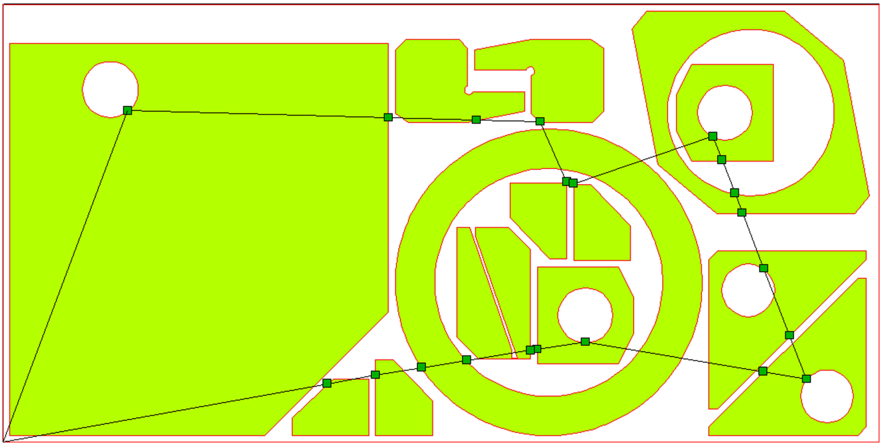


Fig. 8. Solution of CCP, Job #464

Table 1. Solution quality comparison

Job	#229	#464	#3211	#20205
# of parts	11	14	17	115
# of contours	12	21	22	198
# of GTSP points	491	429	493	3917
$\mathcal{L}_{GTSP}$, m	7.729	4.743	4.557	26.098
$\mathcal{L}_{CCP}$, m	7.727	4.706	4.536	25.987

Figure 9 shows the solution to the CCP problem for large dimension (198 contours). Unlike the previous example, for large-dimensional problems it is much more difficult to evaluate the accuracy of the obtained solution. Nevertheless, a comparison with the results of solving the corresponding task GTSP can also serve as a way of estimation. GTSP is known to be NP-hard even on the Euclidean plane [17]. Although it is clear that the bigger the predefined partial order, the simpler the appropriate GTSP task, dependence of theoretical complexity bounds on the properties of the precedence constraints has not yet been insufficiently investigated. In this regard, we note two papers [7,23]. There are two special types of the precedence constraints, for which polynomial time complexity of the GTSP is proven theoretically. The first type of constraints was introduced by E. Balas [2] for the classic TSP. Efficient exact algorithms for the GTSP with precedence constraints of this type are proposed in recent papers [4,5]. Tours that fulfill constraints of the second type are referred to as quasi- and pseudo-pyramidal. Efficient parameterized algorithms for the GTSP with such precedence constraints are proposed in [15,16]. In view of the above, we can summarize that in the field of algorithmic analysis, the GTSP still remains weakly explored. In particular, the absence of efficient Mixed Integer Linear Program (MILP) models for the GTSP makes it impossible to use modern optimizers like Gurobi [11] for construction lower and upper bounds and examining the heuristic solutions. This issue is also pending.

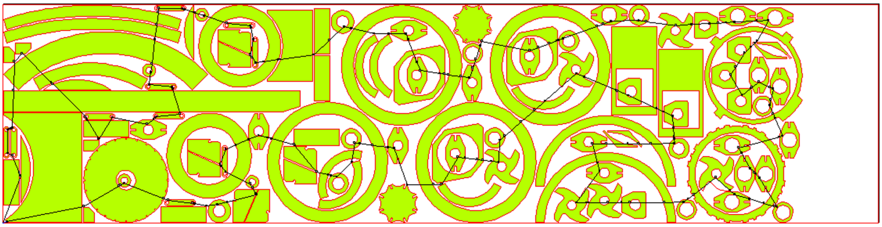


Fig. 9. Example of large problem solution, Job #20205

6 Conclusion

1. The problem of minimizing tool air move of CNC sheet cutting machines for the routing problem from the CCP class is shown to be reduced to a problem without precedence constraint, which reduces the number of contours and the operating time of the algorithm
2. A heuristic algorithm for solving the CCP problem is proposed that does not use contour discretization.
3. It was proved that the CCP-Relax algorithm for finding piercing points for a fixed order of traversing the contours delivers a local minimum.
4. Several easily verified sufficient conditions are formulated and proved for the specified algorithm to deliver global minimum of air move length.
5. CCP-Relax algorithm can be applied to solving more general problems SCCP (Segment Cutting) and GSCCP (Generalized Segment Cutting), thus approaches to solving general ICP cutting problem can be developed on its basis.

The direction of further research is the development of the algorithm for the general case where the pierce points are located outside the contours according to the technological requirements of sheet cutting.

References

1. Arkin, E.M., Hassin, R.: Approximation algorithms for the geometric covering salesman problem. Discret. Appl. Math. **55**(3), 197–218 (1994). https://doi.org/10.1016/0166-218X(94)90008-6

2. Balas, E.: New classes of efficiently solvable generalized traveling salesman problems. Ann. Oper. Res. **86**, 529–558 (1999). https://doi.org/10.1023/A:1018939709890
3. Chentsov, A.G., Chentsov, A.A.: A discrete–continuous routing problem with precedence constraints. Proc. Steklov Inst. Math. **300**(1), 56–71 (2018). https://doi.org/10.1134/S0081543818020074
4. Chentsov, A.G., Khachai, M.Y., Khachai, D.M.: An exact algorithm with linear complexity for a problem of visiting megalopolises. Proc. Steklov Inst. Math. **295**(1), 38–46 (2016). https://doi.org/10.1134/S0081543816090054
5. Chentsov, A., Khachay, M., Khachay, D.: Linear time algorithm for precedence constrained asymmetric generalized traveling salesman problem. IFAC-PapersOnLine **49**(12), 651–655 (2016). https://doi.org/10.1016/j.ifacol.2016.07.767
6. Chentsov, A.G., Chentsov, P.A., Petunin, A.A., Sesekin, A.N.: Model of megalopolises in the tool path optimisation for CNC plate cutting machines. Int. J. Prod. Res. **56**(14), 4819–4830 (2018). https://doi.org/10.1080/00207543.2017.1421784
7. Chentsov, A.G., Grigoryev, A.M.: A scheme of independent calculations in a precedence constrained routing problem. In: Kochetov, Y., Khachay, M., Beresnev, V., Nurminski, E., Pardalos, P. (eds.) DOOR 2016. LNCS, vol. 9869, pp. 121–135. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44914-2_10
8. Dewil, R., Vansteenwegen, P., Cattrysse, D.: Construction heuristics for generating tool paths for laser cutters. Int. J. Prod. Res. **52**(20), 5965–5984 (2014)
9. Dewil, R., Vansteenwegen, P., Cattrysse, D.: A review of cutting path algorithms for laser cutters. Int. J. Adv. Manuf. Technol. **87**(5), 1865–1884 (2016). https://doi.org/10.1007/s00170-016-8609-1
10. Dror, M., Efrat, A., Lubiw, A., Mitchell, J.S.: Touring a sequence of polygons. In: Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing, pp. 473–482. ACM (2003)
11. Gurobi Optimization: Gurobi optimizer reference manual (2020). http://www.gurobi.com
12. Hansen, P., Mladenović, N., Moreno Pérez, J.A.: Variable neighbourhood search: methods and applications. Ann. Oper. Res. **175**(1), 367–407 (2010). https://doi.org/10.1007/s10479-009-0657-6
13. Hoeft, J., Palekar, U.S.: Heuristics for the plate-cutting traveling salesman problem. IIE Trans. **29**(9), 719–731 (1997). https://doi.org/10.1023/A:1018582320737
14. Imahori, S., Kushiya, M., Nakashima, T., Sugihara, K.: Generation of cutter paths for hard material in wire EDM. J. Mater. Process. Technol. **206**(1), 453–461 (2008). https://doi.org/10.1016/j.jmatprotec.2007.12.039
15. Khachai, M.Y., Neznakhina, E.D.: Approximation schemes for the generalized traveling salesman problem. Proc. Steklov Inst. Math. **299**(1), 97–105 (2017). https://doi.org/10.1134/S0081543817090127
16. Khachay, M., Neznakhina, K.: Complexity and approximability of the Euclidean generalized traveling salesman problem in grid clusters. Ann. Math. Artif. Intell. **88**(1), 53–69 (2019). https://doi.org/10.1007/s10472-019-09626-w
17. Papadimitriou, C.H.: Euclidean TSP is NP-complete. Theor. Comput. Sci. **4**, 237–244 (1977)
18. Petunin, A.A., Chentsov, A.A., Chentsov, A.G., Chentsov, P.A.: Elements of dynamic programming in local improvement constructions for heuristic solutions of routing problems with constraints. Autom. Remote Control **78**(4), 666–681 (2017). https://doi.org/10.1134/S0005117917040087

19. Petunin, A.A., Chentsov, A.G., Chentsov, P.A.: About routing in the sheet cutting. Bull. South Ural State Univ. Ser. Math. Model. Program. Comput. Softw. **10**(3), 25–39 (2017). https://doi.org/10.14529/mmp170303
20. Petunin, A.A., Polishchuk, E.G., Ukolov, S.S.: On the new algorithm for solving continuous cutting problem. IFAC-PapersOnLine **52**(13), 2320–2325 (2019). https://doi.org/10.1016/j.ifacol.2019.11.552
21. Petunin, A.: General model of tool path problem for the CNC sheet cutting machines. IFAC-PapersOnLine **52**(13), 2662–2667 (2019)
22. Petunin, A.A., Stylios, C.: Optimization models of tool path problem for CNC sheet metal cutting machines. IFAC-PapersOnLine **49**(12), 23–28 (2016)
23. Saliy, Y.V.: Influence of predestination conditions on the computational complexity of solution of route problems by the dynamic programming method. Bull. Udmurt Univ. Maths. Mech. Comput. Sci. (1), 76–86 (2014)
24. Sherif, S.U., Jawahar, N., Balamurali, M.: Sequential optimization approach for nesting and cutting sequence in laser cutting. J. Manuf. Syst. **33**(4), 624–638 (2014)
25. Smith, S.L., Imeson, F.: GLNS: an effective large neighborhood search heuristic for the generalized traveling salesman problem. Comput. Oper. Res. **87**, 1–19 (2017). https://doi.org/10.1016/j.cor.2017.05.010
26. Vicencio, K., Davis, B., Gentilini, I.: Multi-goal path planning based on the generalized Traveling Salesman Problem with neighborhoods. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2985–2990 (2014). https://doi.org/10.1109/IROS.2014.6942974
27. Ye, J., Chen, Z.G.: An optimized algorithm of numerical cutting-path control in garment manufacturing. Adv. Mater. Res. **796**, 454–457 (2013). https://doi.org/10.4028/www.scientific.net/AMR.796.454
28. Yu, W., Lu, L.: A route planning strategy for the automatic garment cutter based on genetic algorithm. In: 2014 IEEE Congress on Evolutionary Computation (CEC), pp. 379–386 (2014). https://doi.org/10.1109/CEC.2014.6900425