District Division in Arc-Routing

Jiří Fink[1][0000−0001−5065−1213], Vahideh Keikha[2][0000−0003−2821−5903], and Martin Loebl[2][0000−0001−7968−0376]

[1] Department of Theoretical Computer Science and Mathematical Logic, Charles University, Czech Republic
fink@ktiml.mff.cuni.cz
[2] Department of Applied Mathematics, Charles University, Czech Republic
{loebl,keikha}@kam.mff.cuni.cz

Abstract. We model and study winter road maintenance when better maintaining vehicles are used which allows single passing maintenance *in an arbitrary direction* for most of the roads. Hence, the basic building block of a route is an orientation of a cycle. Historically, the maintenance is regionally divided into many independent parts-districts. Each district is a connected part of the regional network, and the regions, aside from being connected, have a geographically local character given by the planarity of the whole network. Such a local division is often required by the regional authorities. This work aims to define such a district partition and suggest algorithms to construct it.

Keywords: Capacity routing · Chinese postman · Arc routing · Graph packing · Matching Theory · Augmentation

1 Introduction

This is a follow-up to our paper [12] where the arc-routing problem (ARP) for *symmetrically oriented networks* is studied. In ARP, the objective is usually defined in terms of minimum cost traversal of a specified arc subset of a graph, possibly with constraints (see, e.g., [6]). For symmetrically oriented networks, the basic building block of a route is the oppositely oriented double edge and the subgraph spanned by a vehicle route is typically a *tree*. This leads to *graph-cutting* and *tree-cutting* problems, see [12].

This current work reflects our experience with the planning of winter road maintenance when better maintenance vehicles are available. These vehicles have more advanced technology which allows them to treat both sides of most of the roads with one passing only. Hence, some wide roads need to be maintained in both directions, but for most of the roads, single passing maintenance *in an arbitrary direction* is sufficient. The most general situation contains one-way streets as well. This changes the problem: the basic building block of a route is *an orientation of a cycle*.

The next important notion of this work is *planarity*. The road networks for the winter road maintenance are essentially planar, with a fixed planar embedding: this is intuitive and was recently tested and discussed in [3] where it is noted that the road networks could be made planar by a small number of adjustments which do not essentially change the practical application.

1.1 Planar aspects of the winter road maintenance

A typical plan of winter road maintenance in a region consists of hundreds of routes and is divided into many independent parts-districts. Each district has a compact local character given by the *planarity of the whole network* and contains one depot. Such a local division

Table 1. The used notations in the paper

Notation	Purpose
$\mathcal{D}$	Depot set
V	Vertex set
A	Arc set
E	Edge set
G	Undirected graph
D	Directed graph
X	Mixed graph
H	Eligible embedded planar graph
G'	Eligible enhanced graph
$\mathcal{B}$	Black faces set
$e(a)$	An inserted arc
$l(a)$	Length of arc a
$c(v)$	Capacity of vertex v
C	Capacity of depots
a, a^{-1}	Opposite oriented arcs
$\mathcal{E}$	Union of the routes
A^a	Set of arcs of $\mathcal{E}$
A^s	One way street set
A^d	Oppositely oriented arcs set
E^d	Set of all double edges
J	Min T-join
J'	Set of edges of min T-join

is often required by the regional authorities. This aspect is important: it makes sense from the maintenance point of view since the winter conditions are similar locally, and also such district divisions reflect the habitual administrative divisions. On the other hand, historically fixed district divisions usually do not take into account the optimization of the winter road maintenance and other arc routing applications of the whole region. This suggests a natural problem to optimize district divisions *around fixed depots*; it is expensive to move a depot and region authorities typically do not want to do that. We refer to this problem as *districting problem*.

1.2 Main contribution

Most previous work for districting problem are heuristics based on plans for minimizing the total maintenance costs, see section 1.3. Our main contribution is to *define* a model of districting, i.e., of a partition of the whole road network into local districts, which involves two parameters crucial for many arc routing applications. Then, we reduce the defined *districting problem* to a novel planar *graph packing problem*.

This connects districting for arc routing to Matching Theory, one of the most developed fields of Discrete Mathematics and Discrete Optimisation.

We then study our approach numerically. A thorough algorithmic and complexity study of the new packing problem is left for future work.

1.3 State of the Art

In [12], we studied the problem of improving the routing for winter road maintenance in some regions in the Czech Republic. We found a formalization of the winter road maintenance based on graph cutting and splitting of necklaces, and our approach was based on integer and linear programming machinery. Our approach is also successfully implemented for winter road maintenance. We note that a recent paper [3] is devoted to exploring the structure of the graph, namely the planarity and bounded degrees, and compares various algorithms with efficiency in mind.

The authors in [4] designed heuristics for districting based on the amalgamation of edges.

In [26], the authors present heuristics for districting which is not based on routing, but on indirectly related composing districts from facial cycles based on *normalized average distances of a face to depots*.

The approach of designing heuristics to actual routing based on facial cycles was introduced in [22] and used later also in [9].

It is argued in [26], based on [10], that actual routing where each route consists of a disjoint union of facial cycles is too rigid to lead to optimal design of actual routes in some situations.

A closely related problem to ARP is the Mixed Chinese Postman Problem (CPP) [14] which is a generalization of the Eulerian path problem. In this problem, we are allowed to traverse each edge more than once. In the following, we explain the problem and its literature in detail.

Given a strongly connected mixed graph $X = (V, E, A)$ and a weight function $w : E \cup A \to N$. The objective is finding a closed tour of X that traverses each edge and arc at least once, of minimum total weight. If X is an undirected graph, then CPP is solved by adding to it a minimum T-join [7].

There is also an efficient algorithm for CPP in strongly connected digraphs, see [8]. CPP becomes NP-complete for generally mixed graphs, see [30].

Since CPP connects the theoretical material with the practical applications, it is extensively studied in the literature. A notable result is that the mixed CPP is fixed-parameter tractable, where the parameter is the number of the arcs [16]. This is an important result since road network graphs in Europe have typically only a few one-way streets. See also [15] and the references therein. We note that several sophisticated heuristics for solving CPP have already been proposed in the literature [11]. The rural postman problem (RPP) is also another variant of the CPP problem. The aim is finding a minimum cost closed route which traverses each arc in a given subset at least once. See [28, 29] for more discussion.

A constrained version of the CPP is the capacitated arc routing problem (CARP) [13] [3]. In this problem, each edge has a non-negative weight and all arcs with positive weight must be traversed by a fleet of identical vehicles of a specific capacity and based at the depots. The CARP problem is also shown to be NP-hard [19]. Different heuristics have been proposed over the years to solve this problem, which include multi-agent methods and Evolutionary algorithms [1, 25, 32, 23, 27, 17], local search [2], Simulated Annealing [9], Tabu search [24] and hybrid methods [20, 18, 31, 5], etc.

In our approach, the basic building block of a district is an *orientation of a facial cycle* of the road network graph. This facilitates the desired local character of the districts and also corresponds to modern trends in winter road maintenance.

The such a basic approach is adopted also in [26] which contains a thorough discussion on principles of districting for arc routing and surveys the previous work. Designing routes is an operational task, the routes can change frequently while districting is a strategic task:

[3] Also known as vehicle routing or dispatch problem.

it is costly and administratively difficult to change district division and thus the districts are defined for an extensive period. Hence, districting needs to be done with a broader view of all arc routing applications as well as the demand for the compact local character of each district.

Our approach is novel: we build districts from faces *using optimization of simplified routing* of the whole region road network graph. The simplified routes are constrained by two parameters as commonly used for most arc-routing applications: the road length enabling to consider minimisation of non-maintaining parts of routes (called dead mileage) and the *capacity* abstracting capacity of maintaining vehicles and robustness of districts.

2 Basic Notions, Definitions and Preprocessing

We consider finite graphs and use standard notations of the graph theory. In particular: (1) $G = (V, E)$ always denotes an undirected graph with the vertex-set V and the edge-set E, (2) $D = (V, A)$ denotes a directed graph where A is the set of its arcs, i.e., ordered pairs of vertices, and (3) $X(V, E, A)$ denotes a mixed graph with the edge-set E and the arc-set A. We say that a mixed graph is *strongly connected* if there is a route from any vertex x to any vertex y, which respects the directions of all arcs it traverses. We will fix a subset $\mathcal{D} \subset V$ of vertices as the set of *depots*. See Table 1 for all notations we used throughout the paper.

We note that to compose districts from orientations of facial cycles, the regional network graph needs to satisfy properties that most network graphs do not meet. Hence, we first *enhance* network graphs by adding arcs and orienting edges and then construct regions using the enhanced digraph. Let us follow the following reasoning:

- Assuming each simplified route is a disjoint union of facial cycles with a prescribed orientation, let $\mathcal{E}$ denote the disjoint union of the routes.
- Let A^a denote the set of the arcs of $\mathcal{E}$, where each arc can appear several times.
- The digraph (V, A^a) has properties that are summarised in the definition of an *eligible network digraph* below.

Definition 1. *Let $D = (V, A)$ be a planar embedded digraph with possibly multiple arcs and with two weight functions on the arc-set, $l : A \to N$ called* length *and $c : A \to N$ called* capacity *. We say that D is* eligible *if it satisfies the following properties:*

- *Each face of the underlying graph of D is bounded by a cycle,*
- *The dual graph D^* is bipartite. Let f^* denote the outer face of D. Faces of D correspond to vertices of D^* and we call* black *the faces of D with corresponding dual vertices in the part of the bi-partition of D^* not containing the vertex corresponding to f^*.*
- *the black faces of D are directed.*

In this section, we will construct, given a (mixed) graph X of a road network, its eligible *enhanced digraph* D^X. The next sections are studying districting of eligible digraphs.

In the following, we recall the definitions used in other parts of the paper.

Minimum Weight Perfect Matching Let $G = (V, E)$ be an undirected graph, where the number of vertices of G is even. A perfect matching is a set of edges that meets each vertex exactly once. For a weighted graph G, the minimum weight perfect matching is a matching with the sum of the weights of its edges minimised.

Eulerian Tour An *Eulerian tour* in a graph (digraph respectively) is a closed tour of the edges (arcs respectively) which contains each edge (arc respectively) exactly once. A graph (digraph respectively) is called *Eulerian* if it admits an Eulerian tour, and it is well-known that a graph is Eulerian if and only if it is connected and all the degrees are even. Also, a digraph is Eulerian if the underlying graph is connected and the in-degree of each vertex equals its out-degree. Another basic fact simple to observe is that the edge-set of each Eulerian graph (digraph respectively) consists of edge-disjoint cycles (directed cycles respectively). Analogously, one defines an Eulerian tour in mixed graphs.

T-joins Given a graph, let T be its set of vertices of an odd degree. A *T-join* is a set of edges of the graph that induces an odd degree in each vertex of T and an even degree (possibly zero) in each vertex not in T. Doubling each edge of a T-join results in a graph where each degree is even.

Minimum T-joins are extensively studied. It is well known that finding a minimum weight T-join in a graph is polynomially reducible to the minimum weight perfect matching. However, with graphs having thousands of vertices, this is still a non-trivial task computationally. We remark that the reductions of minimum T-join to minimum weighted perfect matching are quite old and are being discovered independently in various settings. Perhaps the first such constructions appear in the work of statistical physicists Fisher and Kastelein from the beginning of the 1960s, and Fisher's construction also keeps the planarity of the original graph.

2.1 *Construction of Enhanced Eligible Digraph*

If D is an eligible planar digraph then the black faces can be composed into a closed directed tour containing each arc exactly once. Such tours are Eulerian tours.

Some regional road network mixed graphs are not Eulerian. There is a standard process of enhancing them into Eulerian (di)graphs.

Enhancement Procedure We note that we require more than the Eulerian property from eligible digraphs, namely, we want that each black face is directed. This makes the enhancement procedure more complicated. The following steps obtain an enhanced eligible digraph.

INPUT: We assume that the road network planar mixed graph is $X = (V, E, A)$ where V is the set of vertices, E is the set of the edges and $A = A^d \cup A^s$ is the set of arcs partitioned into the set A^d of *pairs of oppositely oriented arcs* that represent roads maintained in both directions and the set A^s of arcs representing one-way-streets. Typically, A^s is a small set. We further assume X is strongly connected. We are also given two weight functions, the length, and the capacity, on edges and arcs. If a, a^{-1} are oppositely oriented arcs of A^d then we assume that $l(a) = l(a^{-1})$.

Step 1. We update X so that for each $(x, y) \in A^s$ we delete (x, y) from A^s and we add the pair (x, y) and the opposite arc (y, x) to A. We let $c(y, x) = 0$ and $l(y, x)$ be equal to the shortest route in X from y to x. After Step 1, A^s is thus empty and $X = (V, E, A)$ where A consists of double edges only.

Step 2. Let $X = (V, E, A)$ be the output mixed graph after Step 1. Let $G(X)$ be the graph obtained from (V, E) by adding two parallel edges $e_1 = e(a), e_2 = e(a^{-1})$ for each double arc a, a^{-1} of A. For $i = 1, 2$, we let $c(e_1) = c(a)$, $c(e_2) = c(a^{-1})$ and both $l(e_i)$ equal to maximum of $l(a), l(a^{-1})$ because of the following reasoning: if $l(a) \neq l(a^{-1})$ then one of them was added in Step 1 and contributes to dead mileage only. Setting of $l(e_i)$ large will negatively influence frequency of using e_1, e_2 for dead mileage which is preferable around one-way roads.

Each such pair of edges of $G(X)$ will be called a *double edge* and let E^d denote the set of all double edges of $G(X)$.

Step 3. Let J be a minimum T-join of $G(X)$ with respect to edge-weights l.

Step 4. We let $G' = (V, E')$, where E' is the disjoint union of E and J. G' is an embedded planar graph and each degree of G' is even. G' is thus 2-edge-connected, each face of G' is bounded by a cycle and the dual graph of G' is bipartite. Let $\mathcal{B}$ be the part not containing the outer face and let us color the faces of $\mathcal{B}$ with *black*.

Observation 1 *It is possible to orient the edges of G' so that each black face is directed and each double edge is directed oppositely.*

Proof. We let $G'' = (V, E'')$, where $E'' = E' \setminus E^d$. Clearly, G'' is an embedded planar graph and each degree of G' is even and as for G', its dual graph is bipartite. Let $\mathcal{B}''$ be the part not containing the outer face. We orient each face of $\mathcal{B}''$.

It remains to orient the edges of E^d oppositely and according to the original orientation. If a, a^{-1} is such an orientation of a double edge, then there are two possibilities according to where a, a^{-1} is embedded with respect to the faces of G'': inside a face of $\mathcal{B}''$ or inside a face of G'' not in $\mathcal{B}''$. In both case, $\mathcal{B}''$ can be easily modified to yield orientations of faces of $\mathcal{B}$. $\square$

This finishes the enhancement procedure.

3 Districting Eligible Digraphs

From now on, let $H = (V, A)$ be an eligible embedded planar digraph (see Definition 1), let $\mathcal{D} \subset V$ be the set of depots, let $\mathcal{B}$ be the set of the oriented black faces of H and let H be equipped with a length function l and a capacity function c on the arc set.

Districting of H is done as follows: we first do Capacity Routing described in section 3.1. The output is a collection of arc-disjoint routs $R_1, \ldots, R_n$ such that each route consists of a subset of $\mathcal{B}$ and is associated with one vertex-depot of D. This leads to a partition of H into districts so that each district consists of the arcs of the routes associated with the same vertex-depot. In the next section, we define and study Capacity Routing.

3.1 Capacity Routing (CR)

Capacity Routing is an optimization problem with the input consisting of (1) $H = (V, A), D$, $\mathcal{B}$, (2) weight functions l, c and (3) a constant C. It aims to find a partition of A into Eulerian subsets $S_1, \ldots, S_n$ so that (a) each S_i is a union of faces of $\mathcal{B}$ and (b) the total capacity of each S_i is at most C.

We further want that the partition is *optimal*, i.e., the sum of distances (given by length l of the shortest directed path) of the computed districts to D is minimized.

We remark that a variant of the optimization criterion is that the length of both shortest paths to and from a depot is minimized.

3.2 A Reduction to a Packing Problem

We introduce a graphic variant of the Bin Packing Problem which we call the *Plane Bin Packing Problem* (PBPP) and show in Theorem 1 that CR can be reduced to it.

Definition 2 (Plane Bin Packing Problem(PBPP)). *The input consists of (1) connected planar graph $G = (V, E)$, (2) a subset $\mathcal{D}(G) \subset V$, (3) a non-negative integer capacity $c(v)$ for each $v \in V$, (4) a non-negative integer constant C, (5) a non-negative integer length $l(e)$ for each $e \in E$.*

We want to find a partition $V = V_1 \cup \ldots V_m$ so that for each $i \leq m$ (a) V_i induces a connected subgraph and (b) $\sum_{v \in V_i} c(v) \leq C$. We further want that the partition is optimal*, i.e.,*

$$\sum_{i \leq m} L_i + dist(V_i, \mathcal{D}(G))$$

is minimized, where (1) L_i denote the minimum total length of a spanning tree of V_i, and (2) the distance of two subsets of vertices $dist(X, Y)$ is defined as $\min_{x \in X. y \in Y} p(x, y)$ where $p(x, y)$ denotes the length of a shortest path between x, y.

If we do not insist on the planarity of the graph G in Definition 2, we get a more general problem which we call the *Graphic Bin Packing Problem* (GBPP).

Theorem 1. *CR is polynomially reducible to PBPP.*

Proof. Given an input $(H, D, \mathcal{B}, l, c, C)$ of CR, we first contract edges incident to a vertex of degree 2 and split vertices to make sure that each vertex of H is incident with two faces of $\mathcal{B}$. Having this, we construct the input $G = (V(G), E(G))$ of PBPP as follows.

- We let $V(G) = V(H) \cup \{v_f : f \in \mathcal{B}\}$,
- $E(G) = E(H) \cup \{e(f, v) = \{v_f, v\} : v \in V(H) \cap f, f \in \mathcal{B}\}$.
- We note that G is planar,
- We let $c(v_f)$ be equal to the sum of capacities of edges of f in H,
- We let vertex v_f of G belong to set $D(G)$ if f contains a vertex of D. Also, the constant C for G is equal to the constant C of the input of CR.
- It remains to define the length l for G. The lengths $l(e), e \in E(H)$ are inherited from H and we let the lengths of the remaining edges be all equal to zero.

This finishes the construction of the input graph G for PBPP. A solution of PBPP for G clearly translates to a solution of CR for H.

$\square$

4 Graphic Bin Packing Problem

We note that GBPP is a weighted graph packing problem and as such belongs to the *Matching Theory*, a most developed part of discrete optimization.

Theorem 2. *GBPP is polynomially solvable for $C = 2$.*

Proof. We assume that $c(v) = 1$ for each vertex $v \in V$ since otherwise (1) if there exists a vertex v with $c(v) > 2$, then there is no feasible solution, (2) If $c(v) = 2$, then v must form a single-vertex-part of any solution and we can thus delete v from G and consider the smaller problem.

We construct new graph G' with vertex-set disjoint union of V and its copy V', and the edge-set $E \cup \{\{v, v'\}; v \in V\} \cup \binom{V'}{2})$, where v' denotes the copy of $v \in V$ from V'.

We define edge-weights w in G' as follows: (1) If $e \in \binom{V'}{2}$ then $w(e) = 0$, (2) if $e = \{v, v'\}$ then $w(e) = dist(v, D)$ and (3) if $e \in E$ then $w(e) = dist(e, D)$.

We observe that there is a weight-preserving bijection between the set of perfect matchings of G' and feasible solutions of GBPP on G. In particular, the minimum-weight perfect matching algorithm solves the GBPP on G. □

4.1 Case of $C = 3$

This case is already interesting and we do not know its complexity, even for the planar graphs. The problem can be formulated as the problem to cover the vertices of the input graph by vertex disjoint copies of path P_2 and edges so that the total distance to D is minimized. The unweighted version of this problem has been intensively studied in the past, see e.g. [21].

5 Numerical experiments

In this section, we analyze the efficiency of our algorithm for districting problem on several road networks. For all experiments, we used a desktop PC with an i5-8250U processor, running under Linux at 1.6 GHz with 32GB of RAM. We implemented our algorithm in Julia 1.3.

We used the data set from [3] which is available at the Lancaster University Data Repository[4]. This data set contains 18 CPP instances, with cities selected from Paris, London, and Moscow, and with 1000, 2000, 5000, 10 000, 20 000, and 50 000 vertices for each city. We assumed the input graph from the data set is not directed. For simplicity of implementation, we removed all the loops and parallel edges. None of the data sets are planar.

We tested our technique for the whole data set. The maximum running time for the largest data set was less than 10.1 seconds; see Table 2. The computed solution by our algorithm for city London with 1000 vertices is visualized in Fig. 4. In Table 2, we also present the minimum and maximum number of vehicles we required in our approach. Also, we define the magnitude of a depot as the number of vehicles required in that depot. In Fig. 1,2,3, the Histogram of each city represents the frequency distribution of the number of vehicles in the depots. We have merged the associated information included with each city to draw a single Histogram for that city.

We note that number of districts in each data set is computed based on a formula explained bellow. In the following, we describe the preprocessing and experimental procedures required for our algorithm.

[4] http://www.research.lancs.ac.uk/portal/en/datasets/search.html

Selection of the Depots Each data set of [3] has only one depot which does not serve our purposes. In the following, we elaborate on how we produced depots for each data set.

For a planar graph G of n positioned vertices, and given the number of depots d, we consider a grid of size $\sqrt{d} \times \sqrt{d}$ on the bounding box of G, and count the number of vertices inside each grid cell. We then sort the grid cells based on the number of vertices contained in the grid cells and select the d cells of the highest number of vertices. For each of the selected grid cells, an arbitrary random vertex within the cell is chosen as the depot.

Computing Planar Graphs Each input graph is made planar by removing a minimal set of edges, such that none of the removed edges can be added without violating planarity.

Computing Enhanced Eligible Digraph We follow our algorithm of section 2.1.

Graphic Bin Packing Recall that $\mathcal{D}$ denotes the set of the depots. Let the capacity of a face be the total sum of the capacities of all its edges. First, for each face v which is adjacent to a depot, we create a bin B_v and add v into it.

Throughout the packing procedure, we first pack a black face of heaviest capacity which shares a vertex with packed faces in the bins. We use the worst-fit bin packing, at which the emptiest bin will get the connected black face of heaviest capacity.

If there is not enough space left in the bins or the remaining faces are not connected to any of the packed faces, we open a new bin, add to it a non-packed face of heaviest capacity and proceed as above.

We stop when all the faces are packed. The number of bins determines the number of districts, and the packed faces of each bin determine the associated regions to the corresponding district.

See Table 2 for the summary of our experimental results.

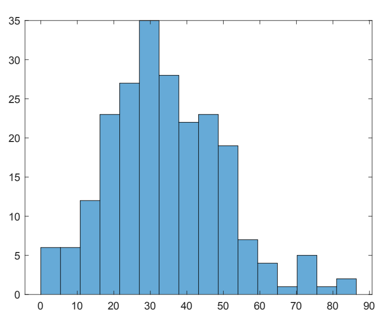

Fig. 1. The magnitude of the depots of the city of London on the data sets of [3].

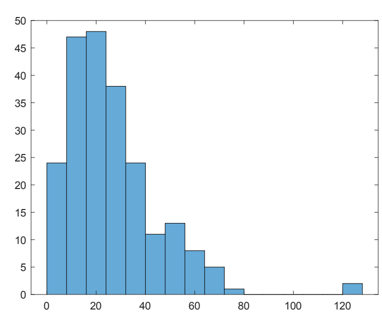

Fig. 2. The magnitude of the depots of the city of Moscow on the data sets of [3].

6 Conclusion

In this paper, we introduced a mathematical model for districting problem, for grouping a set of defined basic geographical units into clusters, such that the total sum of distances of the units to the clusters is optimized. Our practical procedures have shown the efficiency of our approach for real data sets. We note that, although changing the geographical or

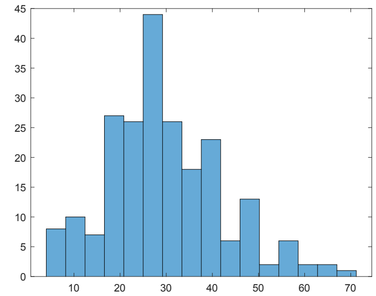

Fig. 3. The magnitude of the depots of the city of Paris on the data sets of [3].

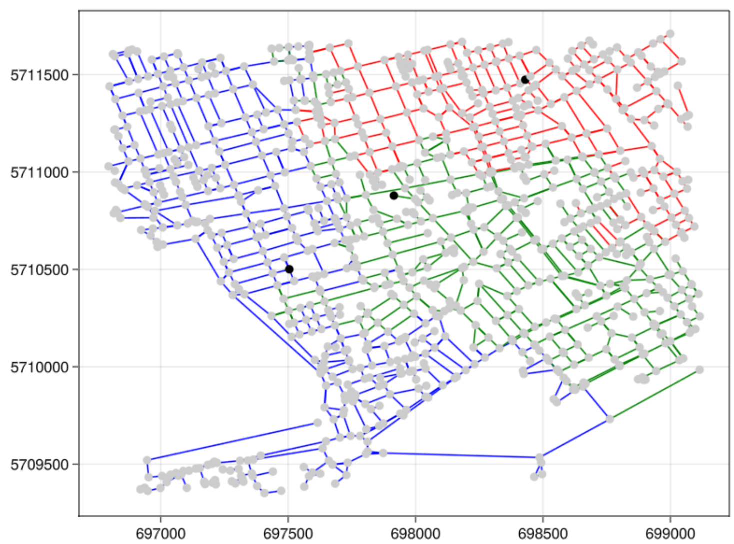

Fig. 4. Three districs of the network of London of 1000 of vertices on the data set of [3]. The depots are highlighted in the given figure.

political districts might not be completely possible in specific case studies, our approach is most valuable at the time when the companies design the sales territories for consumer goods, telecommunications, etc. We finally note that our research stimulates several open problems, e.g., bounding the capacity of the nodes of the road networks, or finding the units of clusters with prescribed combinatorial structures.

Acknowledgement V.Keikha is supported by Charles University project UNCE/SCI/004. J.Fink and M.Loebl are supported by the H2020-MSCA-RISE project CoSP-GA No. 823748.

References

1. J. Bautista, E. Fernández, and J. Pereira. Solving an urban waste collection problem using ants heuristics. *Computers & Operations Research*, 35(9):3020–3033, 2008.
2. P. Beullens, L. Muyldermans, D. Cattrysse, and D. Van Oudheusden. A guided local search heuristic for the capacitated arc routing problem. *European Journal of Operational Research*, 147(3):629–643, 2003.

Districting Problem				
City	Number of vertices	Min number of vehicles per depot	Max number of vehicles per depot	Time (s)
London	1000	3	17	0.0309
	2000	16	33	0.0841
	5000	13	46	0.3199
	10000	13	54	0.6809
	20000	7	65	1.9199
	50000	1	85	8.2175
Moscow	1000	14	21	0.0381
	2000	11	13	0.1597
	5000	4	49	0.2752
	10000	7	71	0.8493
	20000	2	66	2.1436
	50000	1	127	8.5901
Paris	1000	8	33	0.0408
	2000	17	32	0.1419
	5000	14	36	0.2869
	10000	10	40	0.8602
	20000	7	64	2.2409
	50000	4	71	10.085

Table 2. Summary of our experimental results on the data set introduced in [3].

3. B. Boyacı, T. H. Dang, and A. N. Letchford. On matchings, T-joins, and arc routing in road networks. *Networks*, 79(1):20–31, 2022.
4. A. Butsch, J. Kalcsics, and G. Laporte. Districting for arc routing. *INFORMS Journal on Computing*, 26(4):809–824, 2014.
5. B. Cao, W. Zhang, X. Wang, J. Zhao, Y. Gu, and Y. Zhang. A memetic algorithm based on two_arch2 for multi-depot heterogeneous-vehicle capacitated arc routing problem. *Swarm and evolutionary computation*, 63:100864, 2021.
6. Á. Corberán and G. Laporte. *Arc routing: problems, methods, and applications*. SIAM, 2015.
7. J. Edmonds. Chinese postmans problem. In *Operations Research*, page B73. Institute operations research management sciences 901 elkridge landing RD, STE ..., 1965.
8. J. Edmonds and E. L. Johnson. Matching, euler tours and the chinese postman. *Mathematical programming*, 5(1):88–124, 1973.
9. R. Eglese. Routeing winter gritting vehicles. *Discrete applied mathematics*, 48(3):231–244, 1994.
10. R. Eglese and L. Li. Modeling issues in arc routing. *Euro XIII/OR36, Glasgow, Scotland*, 1994.
11. H. A. Eiselt, M. Gendreau, and G. Laporte. Arc routing problems, part ii: The rural postman problem. *Operations research*, 43(3):399–414, 1995.
12. J. Fink, M. Loebl, and P. Pelikánová. Arc-routing for winter road maintenance. *Discrete Optimization*, 41:100644, 2021.
13. B. L. Golden and R. T. Wong. Capacitated arc routing problems. *Networks*, 11(3):305–315, 1981.
14. M. Guan. A survey on the Chinese postman problem. In *Journal of Mathematical Research and Exposition*, volume 4, 1984.
15. G. Gutin, M. Jones, and B. Sheng. Parameterized complexity of the k-arc chinese postman problem. *Journal of Computer and System Sciences*, 84:107–119, 2017.
16. G. Gutin, G. Muciaccia, and A. Yeo. Parameterized complexity of k-Chinese postman problem. *Theoretical Computer Science*, 513:124–128, 2013.
17. V. Q. Hien, T. C. Dao, and H. T. T. Binh. A greedy search based evolutionary algorithm for electric vehicle routing problem. *Applied Intelligence*, pages 1–15, 2022.
18. P. Lacomme, C. Prins, and W. Ramdane-Cherif. Competitive memetic algorithms for arc routing problems. *Annals of Operations Research*, 131(1):159–185, 2004.
19. J. K. Lenstra and A. R. Kan. On general routing problems. *Networks*, 6(3):273–280, 1976.
20. M. Liu and T. Ray. Efficient solution of capacitated arc routing problems with a limited computational budget. In *Proc. Australasian Joint Conference on Artificial Intelligence*, pages 791–802. Springer, 2012.

21. M. Loebl and S. Poljak. On matroids induced by packing subgraphs. *Journal of Combinatorial Theory, Series B*, 44(3):338–354, 1988.
22. J. W. Male and J. C. Liebman. Districting and routing for solid waste collection. *Journal of the Environmental Engineering Division*, 104(1):1–14, 1978.
23. L. d. C. Martins, R. D. Tordecilla, J. Castaneda, A. A. Juan, and J. Faulin. Electric vehicle routing, arc routing, and team orienteering problems in sustainable transportation. *Energies*, 14(16):5131, 2021.
24. Y. Mei, K. Tang, and X. Yao. A global repair operator for capacitated arc routing problem. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(3):723–734, 2009.
25. M. J. Morgan and C. L. Mumford. A weight-coded genetic algorithm for the capacitated arc routing problem. In *Proc. 11th Annual conference on Genetic and evolutionary computation*, pages 325–332, 2009.
26. L. Muyldermans, D. Cattrysse, D. Van Oudheusden, and T. Lotan. Districting for salt spreading operations. *European Journal of Operational Research*, 139(3):521–532, 2002.
27. Z. Nagy, Á. Werner-Stark, and T. Dulai. An artificial bee colony algorithm for static and dynamic capacitated arc routing problems. *Mathematics*, 10(13):2205, 2022.
28. C. Orloff. A fundamental problem in vehicle routing. *Networks*, 4(1):35–64, 1974.
29. C. Orloff. On general routing problems: Comments. *Networks*, 6(3):281–284, 1976.
30. C. H. Papadimitriou. On the complexity of edge traversing. *Journal of the ACM*, 23(3):544–554, 1976.
31. K. Tang, Y. Mei, and X. Yao. Memetic algorithm with extended neighborhood search for capacitated arc routing problems. *IEEE Transactions on Evolutionary Computation*, 13(5):1151–1166, 2009.
32. L. Xing, P. Rohlfshagen, Y. Chen, and X. Yao. An evolutionary approach to the multidepot capacitated arc routing problem. *IEEE Transactions on Evolutionary Computation*, 14(3):356–374, 2009.