View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Kyoto University Research Information Repository
Kyoto University Research Information Repository
KYOTO UNIVERSITY

Title	Network design with edge-connectivity and degree constraints
Author(s)	Fukunaga, Takuro; Nagamochi, Hiroshi
Citation	Theory of Computing Systems (2009), 45(3): 512-532
Issue Date	2009-10
URL	http://hdl.handle.net/2433/87748
Right	c Springer Science+Business Media, LLC 2008.
Type	Journal Article
Textversion	author

Kyoto University

Network design
with edge-connectivity and degree constraints

Takuro Fukunaga Hiroshi Nagamochi

{takuro, nag}@amp.i.kyoto-u.ac.jp
Department of Applied Mathematics and Physics,
Graduate School of Informatics, Kyoto University, Japan

Abstract

We consider the following network design problem; Given a vertex set V with a metric cost c on V, an integer $k \geq 1$, and a degree specification b, find a minimum cost k-edge-connected multigraph on V under the constraint that the degree of each vertex $v \in V$ is equal to $b(v)$. This problem generalizes metric TSP. In this paper, we show that the problem admits a ρ-approximation algorithm if $b(v) \geq 2$, $v \in V$, where $\rho = 2.5$ if k is even, and $\rho = 2.5 + 1.5/k$ if k is odd. We also prove that the digraph version of this problem admits a 2.5-approximation algorithm and discuss some generalization of metric TSP.
Keywords: approximation algorithm, degree constraint, edge-connectivity, (m, n)-VRP, TSP, vehicle routing problem

1 Introduction

It is a main concern in the field of network design to construct a graph of the least cost which satisfies some connectivity requirement. Actually many results on this topic have been obtained so far. In this paper, we consider a network design problem that asks to find a minimum cost k-edge-connected multigraph on a metric edge cost under degree specification. This provides a natural and flexible framework for treating many network design problems. For example, it generalizes the vehicle routing problem with m vehicles (m-VRP) [5, 9], which will be introduced below, and hence contains a well-known metric traveling salesperson problem (TSP), which has already been applied to numerous practical problems [12].

Let $\mathbb{Z}_+$ and $\mathbb{Q}_+$ denote the sets of non-negative integers and non-negative rational numbers, respectively. Let $G = (V, E)$ be a multigraph with a vertex set V and an edge set E, where a multigraph may have some parallel edges but is not allowed to have any loops. For two vertices u and v, an edge joining u and v is denoted by uv. Since we consider multigraphs in this paper, we distinguish two parallel edges $e_1 = uv$ and $e_2 = uv$, which may be simply denoted by uv and uv. For a non-empty vertex set $X \subset V$, $d(X; G)$ (or $d(X)$) denotes the number of edges joining vertices in X and those in $V - X$. In particular $d(v; G)$ (or $d(v)$) denotes the degree of vertex v in G. The edge-connectivity

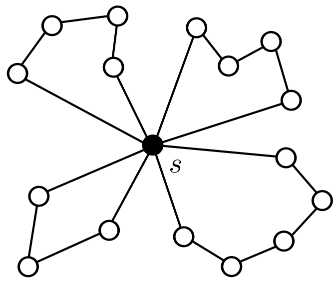

Figure 1: A solution for 4-VRP

$\lambda(u, v; G)$ (or $\lambda(u, v)$) between u and v is the maximum number of edge-disjoint paths between them in G. The edge-connectivity $\lambda(G)$ of G is defined as $\min_{\{u,v\}\in\binom{V}{2}} \lambda(u, v; G)$, where $\binom{V}{2}$ stands for the set of non-ordered pairs of vertices in V. If $\lambda(G) \geq k$ for some $k \in \mathbb{Z}_+$, then G is called *k-edge-connected*. For a function $r : \binom{V}{2} \to \mathbb{Z}_+$, G is called *r-edge-connected* if $\lambda(u, v; G) \geq r(u, v)$ for every $u, v \in V$. For an edge cost $c : \binom{V}{2} \to \mathbb{Q}_+$, $c(E)$ or $c(G)$ denotes $\sum_{e\in E} c(e)$. The edge cost c is called *metric* if it obeys the triangle inequality, i.e., $c(uv) + c(vw) \geq c(uw)$ for every $u, v, w \in V$.

For a degree specification $b : V \to \mathbb{Z}_+$, a multigraph G with $d(v; G) = b(v)$ for all $v \in V$ is called a *perfect b-matching*. In this paper, we focus on the following network design problem.

k-edge-connected multigraph with degree specification (k-ECMDS):
A vertex set V, a metric edge cost $c : \binom{V}{2} \to \mathbb{Q}_+$, a degree specification $b : V \to \mathbb{Z}_+$, and a positive integer k are given. We are asked to find a minimum cost perfect b-matching $G = (V, E)$ of edge-connectivity k. □

In this paper, we suppose that $b(v) \geq 2$ for all $v \in V$ unless stated otherwise, and propose approximation algorithms to k-ECMDS in this case.

Problem k-ECMDS is a generalization of m-VRP, which asks to find a minimum cost set of m cycles, each containing a designated initial city s, such that each of the other cities is covered by exactly one cycle (see Fig. 1). Observe that this problem is 2-ECMDS where $b(s) = 2m$ for the initial city $s \in V$ and $b(v) = 2$ for every $v \in V - s$. If $m = 1$, then m-VRP is exactly TSP. Since TSP is known to be NP-hard [16] even if a given cost is metric (metric TSP), k-ECMDS is also NP-hard. If a given cost is not metric, TSP cannot be approximated unless P = NP [16]. For m-VRP, there is a 2-approximation algorithm based on the primal-dual method [9].

The problem of finding a minimum cost multigraphs subject to either degree or connectivity constraints are well-studied. It is known that finding a minimum cost k-edge-connected graph is NP-hard since it is equivalent to metric TSP when $k = 2$ and a given edge cost is metric. It is 2-approximable by using Jain's algorithm [10] even if solutions are restricted to subgraphs of the given graph and the cost is not metric. On the other hand, it is known that a minimum cost perfect b-matching can be constructed in polynomial time (for example, see [1]). As a prior result on problems equipped with both edge-connectivity requirements and degree constraints, Frank [3] showed that it is polynomially solvable to find a minimum cost r-edge-connected multigraph G with $\ell(v) \leq d(v;G) \leq u(v)$, $v \in V$ for degree lower and upper bounds $\ell, u : V \to \mathbb{Z}_+$ and a metric edge cost c such that $c(uv)$ is defined by $w(u) + w(v)$ for some weight $w : V \to \mathbb{Q}_+$ (in particular, $c(uv) = 1$ for every $uv \in \binom{V}{2}$). Recently Fukunaga and Nagamochi [6] presented approximation algorithms for a network design problem with a general metric edge cost and some degree bounds; For example, they presented a $(2 + 1/\lfloor \min_{u,v \in V} r(u,v)/2 \rfloor)$-approximation algorithm for constructing a minimum cost r-edge-connected multigraph that meets a local-edge-connectivity requirement r with $r(u,v) \geq 2$, $u, v \in V$ under a uniform degree upper bound. Lau et. al. [11] considered the problem to find a minimum cost r-edge-connected subgraph of the given multigraph with degree bounds and general edge cost, and proposed an algorithm which outputs a solution of cost at most twice the optimal although the degree upper bound for a vertex v may be violated up to $2u(v) + 3$. Fukunaga and Nagamochi [7] also gave a 3-approximation algorithm for the case where $r(u,v) \in \{1, 2\}$ for every $u, v \in V$ and $\ell(v) = u(v)$ for each $v \in V$. In this paper, we extend the 3-approximation result [7] to k-ECMDS. Concretely, we prove that k-ECMDS is ρ-approximable if $b(v) \geq 2$, $v \in V$, where $\rho = 2.5$ if k is even and $\rho = 2.5 + 1.5/k$ if k is odd. Moreover, we show that this factor can be improved when a degree specification is uniform.

To design our algorithms for k-ECMDS, we take an approach similar to famous 2- and 1.5-approximation algorithms for metric TSP [16]. These algorithms for metric TSP first construct Eulerian multigraphs, and transform them into Hamiltonian cycles by replacing two edges uv and vz with a new edge uz repeatedly. The edges uv and vz are chosen so that they appear successively in an Eulerian walk of the multigraph. In our algorithms, we first construct the union of a minimum cost perfect b-matching and $\lceil k/2 \rceil$ copies of Hamiltonian cycles constructed by the 1.5-approximation algorithm for metric TSP. We then transform it into a feasible solution by applying the same operation in the algorithms for metric TSP. Two edges to be replaced in the operation are decided based on the structure of the graph in a more sophisticated way than in the algorithms for metric TSP.

We also generalize k-ECMDS to a network design problem in digraphs. We denote an arc (i.e., a directed edge) from a vertex u to another vertex v by uv. Two arcs from u to v are called *parallel*. Let $D = (V, A)$ be a multi-digraph, where a multi-digraph may have some parallel arcs but is not allowed to have any loops. For an ordered pair of vertices u and v, $\lambda(u,v;D)$ (or $\lambda(u,v)$) denotes the arc-connectivity from u to v, i.e., the maximum number of arc-disjoint paths from u to v in D. The arc-connectivity $\lambda(D)$ of D is defined as $\min_{(u,v) \in V \times V} \lambda(u,v;D)$. If $\lambda(D) \geq k$ for some $k \in \mathbb{Z}_+$, D is called *k-arc-connected*. Moreover, $d^-(v;D)$ (or $d^-(v)$) and $d^+(v;D)$ (or $d^+(v)$) denote in- and out-degree of vertex v in digraph D, respectively. Arc cost $c : V \times V \to \mathbb{Q}_+$ is called *symmetric* if $c(uv) = c(vu)$ for every $u, v \in V$, and *metric* if it obeys the triangle inequality, i.e., $c(uv) + c(vz) \geq c(uz)$ for every $u, v, z \in V$.

We call a multi-digraph D with $d^-(v;D) = b^-(v)$ and $d^+(v;D) = b^+(v)$ for all $v \in V$ *perfect (b^-, b^+)-matching* for in- and out-degree specifications $b^-, b^+ : V \to \mathbb{Z}_+$. A minimum cost perfect (b^-, b^+)-matching can be found by computing a minimum cost perfect b-matching in a bipartite graph. The digraph version of the problem is described as follows.

k-arc-connected multi-digraph with degree specification (k-ACMDS):
A vertex set V, a symmetric metric arc cost $c : V \times V \to \mathbb{Q}_+$, in- and out-degree specifications $b^-, b^+ : V \to \mathbb{Z}_+$, and a positive integer k are given. We are asked to find a minimum cost perfect (b^-, b^+)-matching $D = (V, A)$ of arc-connectivity k. □

We also introduce a problem (m, n)-vehicle routing problem ((m, n)-VRP), which generalizes m-VRP so that each city other than a special city is visited by exactly n of the m cycles. Although m-VRP is a special case of k-ECMDS, (m, n)-VRP is not contained in k-ECMDS. However, we show that our algorithm for k-ECMDS also delivers a 2.5-approximate solution to (m, n)-VRP. Moreover, we improve this algorithm to an $(1.5 + (m - n)/m)$-approximation algorithm.

This paper is organized as follows. Section 2 presents an algorithm for the k-ECMDS. Section 3 provides a 2.5-approximation algorithm for the k-ACMDS. Section 4 improves the approximation factors of these algorithms assuming that a degree specification is uniform. Section 5 shows how to apply our algorithm for k-ECMDS to (m, n)-VRP. Section 6 makes some concluding remarks.

2 Algorithm for k-ECMDS

This section describes an approximation algorithm for k-ECMDS. Let (V, b, c, k) be an instance of k-ECMDS. Our algorithm consists of the following three steps.

Feasibility check: The algorithm checks that the given degree specification b satisfies a necessary condition for existence of feasible solutions, which will be found to be also sufficient. If the condition is violated, it outputs message "INFEASIBLE". Otherwise, it goes to the following steps.

Initialization: The algorithm constructs an initial graph whose edge set is the union of $k' = \lceil k/2 \rceil$ Hamiltonian cycles and a perfect b-matching. This initial graph is k-edge-connected by the existence of Hamiltonian cycles. However, it is not feasible because the degree of each $v \in V$ is $b(v) + 2k'$. We will see that its cost is at most 2.5 (resp., $2.5 + 1.5/k$) times the optimal value if k is even (resp., odd).

Transformation into a feasible solution: The algorithm transforms the initial graph into a feasible solution without increasing its cost. This step consists of two phases. Phase 1 modifies only edges in the perfect b-matching while Phase 2 modifies only edges in Hamiltonian cycles.

Now we describe each step one by one. After that, we observe validity of the algorithm.

2.1 Feasibility check

The following theorem describes a necessary and sufficient condition for a degree specification to admit a perfect b-matching. Our algorithm first check the condition and $b(v) \geq k$

for all $v \in V$, which are apparently necessary for an instance to have a k-edge-connected perfect b-matching.

Theorem 1 *Let V be a vertex set with $|V| \geq 2$ and $b : V \to \mathbb{Z}_+$ be a degree specification. Then there exists a perfect b-matching if and only if $\sum_{v \in V} b(v)$ is even and $b(v) \leq \sum_{u \in V-v} b(u)$ for each $v \in V$.*

Proof: The necessity is trivial. We show the sufficiency by constructing a perfect b-matching. We let $V = \{v_1, \ldots, v_n\}$ and $B = \sum_{\ell=1}^{n} b(v_\ell)/2$. For $j = 1, \ldots, B$, we define i_j as the minimum integer such that $\sum_{\ell=1}^{i_j} b(v_\ell) \geq j$, and i'_j as the minimum integer such that $\sum_{\ell=1}^{i'_j} b(v_\ell) \geq B + j$. Notice that $\sum_{\ell=1}^{i_j - 1} b(v_\ell) < j$ holds by the definition if $i_j \geq 2$. Then we can see that $i_j \neq i'_j$ since otherwise we would have $b(v_{i_j}) = \sum_{\ell=1}^{i_j} b(v_\ell) - \sum_{\ell=1}^{i_j - 1} b(v_\ell) > (B + j) - j = B$ if $i_j \geq 2$ and $b(v_{i_j}) \geq B + j > B$ otherwise, which contradicts to the assumption.

Let $M = \{e_j = v_{i_j} v_{i'_j} \mid j = 1, \ldots, B\}$. Then M contains no loop by $i_j \neq i'_j$. Moreover G_M is a perfect b-matching since $|\{j \mid i_j = \ell \text{ or } i'_j = \ell\}| = b(v_\ell)$ for every $\ell = 1, \ldots, n$, as required. $\square$

Note that this theorem is correct even if $b(v) = 1$ for some $v \in V$.

2.2 Initialization

In what follows, we suppose that a perfect b-matching exists and $b(v) \geq k$ for all $v \in V$. If $2 \leq |V| \leq 3$, then every perfect b-matching is k-edge-connected because any non-empty vertex set $X \subset V$ is $\{v\}$ or $V - \{v\}$ for some $v \in V$, and then $d(X) = d(v) \geq k$. Hence we assume without loss of generality that $|V| \geq 4$.

For an edge set F on V, we denote graph (V, F) by G_F. Let M be a minimum cost edge set such that G_M is a perfect b-matching, which is computable in polynomial time [1]. In addition, let H be an edge set of a Hamiltonian cycle on V constructed by the 1.5-approximation algorithm for TSP due to Christofides [16]. In this step, the algorithm prepares M and $k' = \lceil k/2 \rceil$ copies $H_1, \ldots, H_{k'}$ of H. Let E denote the union $M \cup H_1 \cup \cdots \cup H_{k'}$ of them.

Notice that G_E is $2k'$-edge-connected by the existence of edge-disjoint k' Hamiltonian cycles. We call a vertex v in a handling graph G an *excess vertex* if $d(v; G) > b(v)$ (otherwise a *non-excess vertex*). In G_E, all vertices are excess vertices since $d(v; G_E) = b(v) + 2k'$.

2.3 Transformation into a feasible solution

This step reduces the degrees of excess vertices until no excess vertex exists while generating no loops and keeping k-edge-connectivity (Notice that $k < 2k'$ if k is odd). This is achieved by two phases, Phases 1 and 2, as follows.

Phase 1: In this phase, we modify only edges in M while keeping edges in $H_1, \ldots, H_{k'}$ unchanged. We define the following two operations on an excess vertex $v \in V$.

Operation 1: If v has two incident edges xv and yv in M with $x \neq y$, replace xv and yv by new edge xy.

Operation 2: If v has two parallel edges uv in M with $d(u) > b(u)$, remove those two edges.

Phase 1 repeats Operations 1 and 2 until none of them is executable. For avoiding ambiguity, we let M' denote M after executing Phase 1, and M denote the original set in what follows. Moreover, let $E' = M' \cup H_1 \cup \cdots \cup H_{k'}$. Note that $d(v) - b(v)$ is always a non-negative even integer for all $v \in V$ throughout (and after) these operations because $d(v; G_E) - b(v) = 2k'$ and each operation decreases the degree of a vertex by 2. If no excess vertex remains in $G_{E'}$, then we are done. We consider the case in which there remain some excess vertices, and show some properties on M' before describing Phase 2.

Lemma 1 *Every excess vertex in $G_{E'}$ has at least one incident edge in M'. If it has more than one incident edges in M', then they are parallel.*

Proof: Since $d(v; G_{E'}) - b(v)$ is a positive even integer for an excess vertex v in $G_{E'}$, it holds that $d(v; G_{M'}) = d(v; G_{E'}) - d(v; G_{H_1 \cup \cdots \cup H_{k'}}) \geq (b(v) + 2) - 2k' > 0$. Hence v has at least one incident edges in M'. If v has more than one incident edges in M', then they are parallel since otherwise Operation 1 can be applied to v. □

For an excess vertex v in $G_{E'}$, let $n(v)$ denote the unique neighbor of v in $G_{M'}$. If $n(v)$ is also an excess vertex in $G_{E'}$, we call the pair $\{v, n(v)\}$ by a *strict pair*.

Lemma 2 *Let $\{v, n(v)\}$ be a strict pair. Then $d(v; G_{M'}) = d(n(v); G_{M'}) = 1$, k is odd, and $b(v) = b(n(v)) = k$.*

Proof: Since both v and $n(v)$ are excess vertices, only edges between u and v are incident to them in M' by Lemma 1. Hence $d(v; G_{M'}) = d(n(v); G_{M'})$. Moreover, $d(v; G_{M'}) = d(n(v); G_{M'}) = 1$ holds since otherwise Operation 2 can be applied to v and $n(v)$.

Let $u \in \{v, n(v)\}$. Then it holds that $d(u; G_{E'}) = d(u; G_{H_1 \cup \cdots \cup H_{k'}}) + d(u; G_{M'}) = 2k' + 1 = 2\lceil k/2 \rceil + 1$. Since $d(u; G_{E'}) - b(u)$ is even, $b(u)$ must be odd. This fact and $d(u, G_{E'}) > b(u) \geq k$ indicates that $b(u) = k$ and k is odd. □

If v is an excess vertex in no strict pair, $n(v)$ is a non-excess vertex. In other words, the existence of excess vertices in no strict pairs indicates that of some non-excess vertices. Upon completion of Phase 1, let N denote the set of non-excess vertices in $G_{E'}$, and S denote the set of strict pairs in $G_{E'}$. If $N = \emptyset$, all excess vertices are in some strict pairs. By Lemma 2, k is an odd integer in this case, and furthermore $k \geq 3$ by the assumption that $b(v) \geq 2$, $v \in V$ if $k = 1$. From this fact and $|V| \geq 4$, we have the following.

Lemma 3 *If $N = \emptyset$, then $|S| \geq 2$.* □

Phase 2 ($|V| = 4$, $|S| = 2$): Now we describe Phase 2. First, we deal with a special case of $|V| = 4$ and $|S| = 2$.

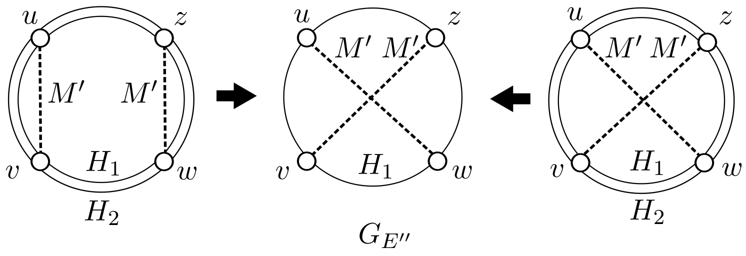

Figure 2: Operations when $|V| = 4$ and $|S| = 2$

Lemma 4 *If $|V| = 4$ and $|S| = 2$ after Phase 1, we can transform $G_{E'}$ into a k-edge-connected perfect b-matching without increasing the cost.*

Proof: Let $V = \{u, v, w, z\}$ and $H = \{uv, vw, wz, zu\}$. Now $E' = M' \cup H_1 \cup \cdots \cup H_{k'}$ ($k \geq 2$). Then either $M' = \{uv, wz\}$ (or $\{vw, zu\}$) or $M' = \{uw, vz\}$ holds. In both cases, we replace $M' \cup H_1 \cup H_2$ by $E'' = \{uv, vw, wz, zu, uw, vz\}$ (see Fig. 2). Then, we can see that $d(x; G_{E''}) = 3$ for all $x \in V$ and $G_{E''}$ is 3-edge-connected. Since $d(x; G_{H_i}) = 2$ for $x \in V, i = 3, \ldots, k'$ and G_{H_i} is 2-edge-connected for $i = 3, \ldots, k'$, it holds that $d(x; G_{E'' \cup H_3 \cup \cdots \cup H_{k'}}) = 3 + 2(k' - 2) = k = b(x)$ for $x \in V$ and the edge-connectivity of $G_{E'' \cup H_3 \cup \cdots \cup H_{k'}}$ is $3 + 2(k' - 2) = k$ (The existence of strict pair implies that k is odd by Lemma 2.).

Hence it suffices to show that $c(E'') \leq c(M') + c(H_1) + c(H_2)$. If $M' = \{uw, vz\}$, then it is obvious since $E'' = M' \cup H_1 \subseteq M' \cup H_1 \cup H_2$. Let us consider the other case, i.e., $M' = \{uv, wz\}$ (or $\{vw, zu\}$). From $M' \cup H_1 \cup H_2$, remove $\{uv, uv\}$, replace $\{wz, zu\}$ by $\{wu\}$, and replace $\{vw, wz\}$ by $\{vz\}$. Then the edge set becomes E'' without increasing edge cost, as required. $\square$

Phase 2 (the other case): In the following, we consider the other case, to which Lemma 4 cannot be applied, i.e., $N \neq \emptyset$ or $|S| \geq 3$. In this case, Phase 2 modifies only edges in H_i, $i = 1, \ldots, k'$ while keeping the edges in M' unchanged. Let $V(H_i)$ denote the set of vertices covered by H_i. We define *detaching v from cycle H_i* to be an operation that replaces the pair $\{uv, vw\} \subseteq H_i$ of edges incident to v by a new edge uw. Note that this decreases $d(v)$ by 2, but H_i remains a cycle on $V(H_i) := V(H_i) - \{v\}$.

Let v be an excess vertex in $G_{E'}$. If v is in a strict pair (i.e., $n(v)$ is also an excess vertex), Phase 2 detaches v from a cycle H_i and $n(v)$ from a cycle H_j such that $v \in V(H_i)$, $n(v) \in V(H_j)$ and $i \neq j$. By this operation, v and $n(v)$ become non-strict vertices since $d(x; G_{E'}) = d(x; G_{H_1 \cup \cdots \cup H_{k'}}) + d(x; G_{M'}) = 2k' + 1 = b(x) + 2$ holds for $x \in \{v, n(v)\}$ by Lemma 2. On the other hand, if v is in no strict pair, Phase 2 reduces $d(v)$ to $b(v)$ by detaching v from $(d(v; G_{E'}) - b(v))/2$ cycles in $H_1, \ldots, H_{k'}$ each of which covers larger number of vertices than the others. Notice that $(d(v; G_{E'}) - b(v))/2 \leq k'$ by $d(v; G_{E'}) - b(v) \leq d(v; G_E) - b(v) = 2k'$. Moreover, observe that $|V(H_i)| \leq |V(H_j)| + 1$ always holds for any $i, j \in \{1, \ldots, k'\}$ during this phase.

In the following, we let H'_i denote H_i after Phase 2. Moreover let $E'' = M' \cup H'_1 \cup \cdots \cup H'_{k'}$. The algorithm outputs $G_{E''}$.

2.4 Validity of our algorithm

The entire of our algorithm is described as follows.

Algorithm UNDIRECT(k)

Input: A vertex set V, a degree specification $b : V \to \mathbb{Z}_+$, a metric edge cost $c : V \to \mathbb{Q}_+$, and a positive integer k

Output: A k-edge-connected perfect b-matching or "INFEASIBLE"

\# Feasibility check
if $\sum_{v \in V} b(v)$ is odd, $\exists v : b(v) > \sum_{u \in V-v} b(u)$ or $k > b(v)$ **then**
  Output "INFEASIBLE" and halt
end if;

\# Initialization
Compute a minimum cost perfect b-matching G_M;
if $|V| \leq 3$ **then**
  Output G_M and halt
end if;
Compute a Hamiltonian cycle G_H on V by Christofides' algorithm;
$k' := \lceil k/2 \rceil$; Let $H_1, \ldots, H_{k'}$ be k' copies of H;

\# Phase 1 of transformation into a feasible solution
$M' := M$;
while Operation 1 or 2 is applicable to a vertex $v \in V$
with $d(v; G_{M' \cup H_1 \cup \cdots \cup H_{k'}}) > b(v)$ **do**
  if $\exists \{xv, vy\} \subseteq M'$ such that $x \neq y$ **then**
    $M' := (M' - \{xv, vy\}) \cup \{xy\}$ \# Operation 1
  else
    if $\exists \{xv, xv\} \subseteq M'$ such that $d(x; G_{M' \cup H_1 \cup \cdots \cup H_{k'}}) > b(x)$ **then**
      $M' := M' - \{xv, vx\}$ \# Operation 2
    end if
  end if
end while;

\# Phase 2 ($|V| = 4$, $|S| = 2$) of transformation into a feasible solution
if V consists of only two strict pairs **then**
  Rename vertices so that $H = \{uv, vw, wz, zu\}$;
  $H'_2 := \emptyset$; $M' := \{uw, vz\}$;
  Output $G_{M' \cup H'_1 \cup \cdots \cup H'_{k'}}$ and halt
end if;

\# Phase 2 (the other case) of transformation into a feasible solution
$H'_i := H_i$ for each $i = 1, \ldots, k'$;
for all $v \in V$ with $d(v; G_{M' \cup H'_1 \cup \cdots \cup H'_{k'}}) > b(v)$ **do**
  if v and $n(v)$ forms a strict pair **then**
    Detach v from any H'_i and $n(v)$ from any H'_j with $i \neq j$

else
  while $d(v; G_{M' \cup H'_1 \cup \cdots \cup H'_{k'}}) > b(v)$ **do**
    Choose $i \in \{1, \ldots, k'\}$ such that $v \in V(H'_i)$ and $|V(H'_i)| \geq |V(H'_j)|$ for all $j \neq i$ with $v \in V(H'_j)$;
    Detach v from H'_i
  end while
end if
end for;
$E'' := M' \cup H'_1 \cup \cdots \cup H'_{k'}$;
Output $G_{E''}$

The feasibility check takes running time of $O(|V|)$. The Phase 1 of transformation into a feasible solution takes running time of $O(|V|^3)$. Running time of Phase 2 seems to depend on k. However, this phase can be implemented in $O(|V|^2)$. Hence the running time of the entire algorithm is $O(|V|^3 + h + m)$, where h is the running time of the Christofides' algorithm and m is that of computing a minimum cost perfect b-matching.

The following two lemmas states the feasibility of $G_{E''}$.

Lemma 5 *$G_{E''}$ is a perfect b-matching.*

Proof: We have already seen the case of $|V| = 4$ and $|S| = 2$ in Lemma 4. Hence we suppose the other case here. In the following, we show that Phase 2 in the algorithm keeps $V(H_i) \geq 2$ for $i = 1, \ldots, k'$. This means that the algorithm transform G_E into a perfect b-matching without generating any loop.

First, let us consider the case of $S \neq \emptyset$. Recall $k \geq 3$ and $k' = \lceil k/2 \rceil \geq 2$ in this case. For each strict pair $\{u, v\} \in S$, Phase 2 detaches u and v from different cycles in $H_1, \ldots, H_{k'}$. After this, each of $H_1, \ldots, H_{k'}$ is incident to at least one vertex of any strict pair in S in addition to all vertices in N. Since $|S| = 1$ implies $|N| > 0$ by the assumption, it holds that $|V(H_i)| \geq |S| + |N| \geq 2$ for each $i = 1, \ldots, k'$, as required.

Next, let us consider the case of $S = \emptyset$. In this case, $|N| \geq 1$ holds because the existence of an excess vertex in no strict pair implies that of a non-excess vertex. If $|N| \geq 2$, the claim is obvious since each of $H_1, \ldots, H_{k'}$ is always incident to all vertices in N. Hence suppose that $|N| = 1$, and let x be the vertex in N. All edges in M' are incident to x since otherwise $S = \emptyset$ implies that Operation 1 or 2 would be applicable to some vertex in $V - x$. In other words, $b(x) = d(x; G_{E'}) = |M'| + 2k'$ holds before Phase 2. Moreover $\sum_{v \in V - x} b(v) \geq b(x)$ also holds by the assumption that b admits the existence of perfect b-matchings.

Now assume that we have converted some excess vertices in $G_{E'}$ into non-excess vertices by detaching them from some of $H_1, \ldots, H_{k'}$ while keeping $|V(H_i)| \geq 2$, $i = 1, \ldots, k'$, and yet an excess vertex $y \in V - x$ remains. Let H_{i_1} be a cycle covering y (such H_{i_1} always exists). If $|V(H_{i_1})| \geq 3$, then we are done. Hence let us suppose the otherwise (i.e., $|V(H_{i_1})| = 2$).

The existence of y implies that $\sum_{v \in V} d(v) > \sum_{v \in V} b(v)$. Then there remains a cycle

H_{i_2} with $|V(H_{i_2})| \geq 3$ because

$$2 \sum_{1 \leq j \leq k'} |V(H_j)| = \sum_{v \in V} d(v; G_{H_1 \cup \cdots \cup H_{k'}}) = \sum_{v \in V} d(v) - 2|M'|$$
$$> \sum_{v \in V - \{x\}} b(v) + b(x) - 2|M'| \geq 2(b(x) - |M'|) = 4k'.$$

Let $y \notin V(H_{i_2})$, i.e., y has already been detached from H_{i_2}. Before y is detached from H_{i_2}, it holds that $|V(H_{i_2})| \geq 4$. Hence $|V(H_{i_2})| > |V(H_{i_1})| + 1$ holds at this moment. However the algorithm preserves property $|V(H_i)| \leq |V(H_j)| + 1$ for any $i, j \in \{1, \ldots, k'\}$ as stated above. Therefore we have a contradiction. □

Lemma 6 *$G_{E''}$ is k-edge-connected.*

Proof: We have already seen the case of $|V| = 4$ and $|S| = 2$ in Lemma 4. Hence we suppose the other case here. Let $u, v \in V$. We show that $\lambda(u, v; G_{E''}) \geq k$.

Case-1: First suppose that u and v are in some (possibly different) strict pairs in $G_{E'}$. Moreover, let $u \notin V(H'_i)$ and $v \notin V(H'_j)$ (hence $u \in V(H'_\ell)$ for all $\ell \in \{1, \ldots, k'\} - \{i\}$ and $v \in V(H'_\ell)$ for all $\ell \in \{1, \ldots, k'\} - \{j\}$). For each $\ell \in \{1, \ldots, k'\} - \{i, j\}$, $\lambda(u, v; G_{H'_\ell}) = 2$ holds because $u, v \in V(H'_\ell)$. If $i = j$, $\lambda(u, v; G_{H'_i \cup M'}) = 1$ holds because $d(u; G_{M'}) = d(v; G_{M'}) = 1$ and $n(u), n(v) \in V(H'_i)$. Then it holds that $\lambda(u, v; G_{E''}) = 2(k' - 1) + 1 = k$ in this case (Recall that the existence of strict pairs implies that k is odd by Lemma 2). Hence we let $i \neq j$, and show that $\lambda(u, v; G_{H'_i \cup H'_j \cup M'}) \geq 3$ from now on, from which $\lambda(u, v; G_{E''}) \geq 2(k' - 2) + 3 = k$ can be derived.

Suppose that $V(H'_i) \cap V(H'_j) = \emptyset$. In this case, it can be seen that $N = \emptyset$, and hence $|S| \geq 3$ by the assumption about the relationship between N and S. Since at least one vertex of each strict pair is covered by each cycle in $H'_1, \ldots, H'_{k'}$, we can see that M' contains at least three vertex-disjoint edges that join vertices in $V(H'_i)$ and in $V(H'_j)$, two of which are u and v. This indicates that $\lambda(u, v; G_{H'_i \cup H'_j \cup M'}) \geq 3$ holds (see the graph of Figure 3 (b)).

Let us consider the case of $V(H'_i) \cap V(H'_j) \neq \emptyset$ next. By the existence of u and v, $|S| \geq 1$ holds. If u and v forms a strict pair (i.e., $uv \in M'$), $\lambda(u, v; G_{M'}) = 1$ holds. Since $V(H'_i) \cap V(H'_j) \neq \emptyset$ implies $\lambda(G_{H'_i \cup H'_j}) \geq 2$, we see that $\lambda(u, v; G_{H'_i \cup H'_j \cup M'}) \geq 3$ in this case. Thus let u and v belong to different strict pairs (i.e., $|S| \geq 2$). Then there exists two vertex-disjoint edges in M' joins vertices in $V(H'_i)$ and in $V(H'_j)$ (see Figure 3 (a)). If we split each vertex $w \in V(H'_i) \cap V(H'_j)$ into two vertices w' and w'' so that H'_i and H'_j are vertex-disjoint cycles, and add new edges $w'w''$ joining those two split vertices to M', then we can reduce this case to the case of $V(H'_i) \cap V(H'_j) = \emptyset$, in which $\lambda(u, v; G_{H'_i \cup H'_j \cup M'}) \geq 3$ has already been observed in the above (see Figure 3). Accordingly, we have $\lambda(u, v; G_{H'_i \cup H'_j \cup M'}) \geq 3$ if u and v are in some strict pairs, as required.

Case-2: In the next, let both u and v be not in any strict pairs. For $z \in \{u, v\}$, let $n'(z)$ denote z itself if $z \in N$, and $n(z)$ otherwise. Notice that $n'(z) \in N$ for any $z \in \{u, v\}$, i.e., it is covered by all of $H'_1, \ldots, H'_{k'}$. If $z \in \{u, v\}$ is not covered by $p > 0$ cycles in $H'_1, \ldots, H'_{k'}$ (and hence z is an excess vertex in $G_{E'}$), then z has at least $k - 2(k' - p)$ incident edges in M' because $d(z; G_{M'}) = b(z) - d(z; G_{H'_1 \cup \cdots \cup H'_{k'}}) \geq k - 2(k' - p)$. Hence

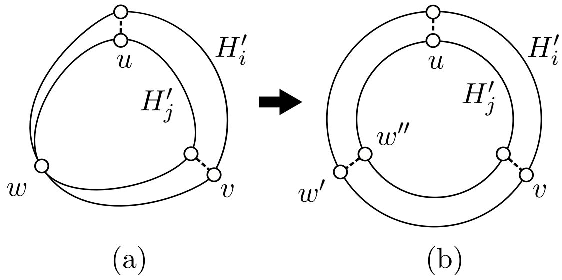

Figure 3: Reduction to the case of $V(H'_i) \cap V(H'_j) = \emptyset$

$\lambda(z, n'(z); G_{E''}) \geq 2(k'-p) + k - 2(k'-p) = k$ holds for each $z \in \{u, v\}$, where we define $\lambda(z, z; G_{E''}) = +\infty$. Moreover it is obvious that $\lambda(n'(u), n'(v); G_{E''}) \geq 2k'$. Therefore, it holds that

$$\lambda(u, v; G_{E''}) \geq \min\{\lambda(u, n'(u); G_{E''}), \lambda(n'(u), n'(v); G_{E''}), \lambda(n'(v), v; G_{E''})\} \geq k.$$

Case-3: Finally, let us consider the remaining case, i.e., u is in a strict pair and v is a vertex which is not in any strict pair. Let us define $n'(v)$ as before. Then $\lambda(v, n'(v); G_{E''}) \geq k$ holds. Without loss of generality, let u be detached from H'_1, and spanned by $H'_2, \ldots, H'_{k'}$. Since $un(u) \in M'$ and $n(u), n'(v) \in V(H'_1)$, it holds that $\lambda(u, n(u); G_{M' \cup H'_1}) = 1$, and $\lambda(n(u), n'(v); G_{M' \cup H'_1}) \geq 2$. Then,

$$\begin{aligned}\lambda(u, n'(v); G_{E''}) \geq \min\{\lambda(u, n(u); G_{M' \cup H'_1}), \lambda(n(u), n'(v); G_{M' \cup H'_1})\} \\ + \lambda(u, n'(v); G_{H'_2 \cup \cdots \cup H'_{k'}}) \geq 1 + 2(k'-1) = 2k' - 1 = k.\end{aligned}$$

Therefore,

$$\lambda(u, v; G_{E''}) \geq \min\{\lambda(u, n'(v); G_{E''}), \lambda(v, n'(v); G_{E''})\} \geq k,$$

holds, as required. □

Let us consider the cost of the graph $G_{E''}$. The following theorem on the Christofides' algorithm gives us an upper bound on $c(H)$. Here, we let $\delta(U)$ denote the set of edges whose one end vertex is in U and the other is in $V - U$ for nonempty $U \subset V$.

Theorem 2 ([8, 17]) *Let*

$$\begin{array}{lll} OPT_{TSP} = \min & \sum_{e \in E} c(e)x(e) & \\ \textit{subject to} & \sum_{e \in \delta(U)} x(e) \geq 2 & \textit{for each nonempty } U \subset V, \\ & x(e) \geq 0 & \textit{for each } e \in E. \end{array}$$

Christofides' algorithm for TSP always outputs a solution of cost at most $1.5 OPT_{TSP}$. □

Lemma 7 $c(E'')$ *is at most* $1 + 3\lceil k/2 \rceil / k$ *times the optimal cost of k-ECMDS.*

Proof: No operation in Phases 1 and 2 increases the cost of the graph since the edge cost is metric. Hence it suffices to show that $c(M \cup H_1 \cup \cdots \cup H_{k'})$ is at most $(1 + 3\lceil k/2 \rceil / k) \cdot c(G)$, where G denotes an optimal solution of k-ECMDS. Since G is a perfect b-matching, $c(M) \leq c(G)$ obviously holds. Thus it suffices to show that $c(H_i) \leq 3c(G)/k$ for $1 \leq i \leq k'$, from which the claim follows.

Let $x_G : \binom{V}{2} \to \mathbb{Z}_+$ be the function such that $x_G(uv)$ denotes the number of edges joining u and v in G. Since G is k-edge-connected, $\sum_{e \in \delta(U)} x_G(e) \geq k$ holds for every nonempty $U \subset V$. Hence $2x_G/k$ is feasible for the linear programming in Theorem 2, which means that $\mathrm{OPT}_{TSP} \leq 2c(G)/k$. By Theorem 2, $c(H_i) \leq 1.5\mathrm{OPT}_{TSP}$. Therefore we have $c(H_i) \leq 3c(G)/k$, as required. $\square$

Lemmas 5, 6 and 7 establish the next.

Theorem 3 *Algorithm* UNDIRECT(k) *is a ρ-approximation algorithm for k-ECMDS, where $\rho = 2.5$ if k is even and $\rho = 2.5 + 1.5/k$ if k is odd.* $\square$

Algorithm UNDIRECT(k) always outputs a solution for $k \geq 2$ as long as there exists a perfect b-matching and $b(v) \geq k$ for all $v \in V$. This fact and Theorem 1 imply the following corollary.

Corollary 1 *For $k \geq 2$, there exists a k-edge-connected perfect b-matching if and only if $\sum_{v \in V} b(v)$ is even and $k \leq b(v) \leq \sum_{u \in V-v} b(u)$ for all $v \in V$.* $\square$

We close this section with a few remarks. The operations in the transformation into a feasible solution are equivalent to a graph transformation called *splitting*, followed by removing generated loops if any. There are many results on the conditions for splitting to maintain the edge-connectivity [4, 13]. In fact, these results are used for designing algorithms computing a minimum cost multigraphs with edge-connectivity and degree constraints [3, 6]. However, we can not use them in our problem because splitting in these results may generate loops. Hence algorithm UNDIRECT(k) needs to specify a sequence of splitting so that removing loops does not make the degrees lower than the degree specification.

One may consider that a perfect $(b-2k')$-matching is more appropriate than a perfect b-matching as a building block of our algorithm, since there is no excess vertex for the union of a perfect $(b-2k')$-matching and k' Hamiltonian cycles. However, there is a degree specification b that admits a perfect b-matching, and no perfect $(b-2k')$-matching (for example, see $V = \{u, v, z\}$, $b(u) = b(v) = 3$, $b(z) = 6$ and $k' = 1$). Furthermore, even if there exits a perfect $(b-2k')$-matching, the minimum cost of perfect $(b-2k')$-matchings may not be a lower bound on the optimal cost of k-ECMDS. Therefore we do not use a perfect $(b-2k')$-matching in general case. In Section 4, we show that a perfect $(b-2k')$-matching always exist and its cost can be estimated when a degree specification b is uniform.

3 Algorithm for k-ACMDS

This section shows that k-ACMDS is 2.5-approximable. The algorithm for k-ACMDS can be designed analogously with that for k-ECMDS. It also consists of feasibility check, initialization, and transformation into a feasible solution. In what follows, let us describe them.

3.1 Feasibility check

Let (V, b^-, b^+, c, k) be an instance of k-ACMDS. In this subsection, we state conditions necessary for the instance to have a feasible solution, which will be found to be also sufficient. The algorithm first check the condition and if it is violated, then output message "INFEASIBLE".

Frobenius' classic theorem (see [14] for example) tells the relationship between the existence of perfect bipartite matchings and the minimum size of vertex covers in bipartite graphs.

Theorem 4 (Frobenius) *A bipartite graph G has a perfect matching if and only if each vertex cover has size at least $|V(G)|/2$.* $\square$

From this, we can immediately derive a condition for a digraph to have a perfect (b^-, b^+)-matching.

Theorem 5 *Let V be a vertex set, and $b^-, b^+ : V \to \mathbb{Z}_+$ be in- and out- degree specifications, respectively. There exists a perfect (b^-, b^+)-matching if and only if $\sum_{v\in V} b^-(v) = \sum_{v\in V} b^+(v)$, $b^-(v) \le \sum_{u\in V-v} b^+(u)$ for each $v \in V$, and $b^+(v) \le \sum_{u\in V-v} b^-(u)$ for each $v \in V$.*

Proof: The necessity is obvious. Hence we consider the sufficiency in the following. For each $v \in V$, prepare two vertex sets V_v^- and V_v^+ corresponding to v such that $|V_v^-| = b^-(v)$ and $|V_v^+| = b^+(v)$. Furthermore, let $V^- = \cup_{v\in V} V_v^-$, $V^+ = \cup_{v\in V} V_v^+$, and $E = \{u^-v^+ \mid u^- \in V_u^-, v^+ \in V_v^+, u \neq v\}$. Then a perfect matching in a bipartite graph (V^-, V^+, E) corresponds to a perfect (b^-, b^+)-matching on V. So by Theorem 4, it suffices to show that each vertex cover of (V^-, V^+, E) has size at least $(|V^-| + |V^+|)/2$.

To the contrary, let us suppose that there exists a vertex cover $C \subset V^- \cup V^+$ of (V^-, V^+, E) such that $|C| < (|V^-|+|V^+|)/2$ under the assumption in this theorem. Since $|V^-| = \sum_{v\in V} b^-(v) = \sum_{v\in V} b^+(v) = |V^+|$, it holds that $|C| < |V^-| = |V^+|$. This implies the existence of vertices $x \in V^- - C$ and $y \in V^+ - C$. Let x correspond to $u \in V$ (i.e., $x \in V_u^-$) and y correspond to $v \in V$ (i.e., $y \in V_v^+$). If $u \neq v$, there exists an edge $xy \in E$, which is not covered by any vertices in C, a contradiction. Hence $u = v$ holds. Then $\cup_{z\in V-v}(V_z^- \cup V_z^+) \subseteq C$ holds. This implies that $|C| \ge \sum_{z\in V-v} |V_z^-| + \sum_{z\in V-v} |V_z^+|$. Then it holds that

$$\begin{aligned}(\sum_{v\in V} b^-(v) + \sum_{v\in V} b^+(v))/2 &= (|V^-| + |V^+|)/2 > |C| \\ &\ge \sum_{z\in V-v} |V_z^-| + \sum_{z\in V-v} |V_z^+| = \sum_{z\in V-v} b^-(z) + \sum_{z\in V-v} b^+(z),\end{aligned}$$

implying $b^-(v) + b^+(v) > \sum_{z\in V-v} b^-(z) + \sum_{z\in V-v} b^+(z)$. However, this indicates that at least $b^-(v) > \sum_{z\in V-v} b^-(z)$ or $b^+(v) > \sum_{z\in V-v} b^+(z)$ holds, contradicting to the assumption. $\square$

The conditions in Theorem 5 is apparently necessary for feasibility of the given instance. In addition, the algorithm checks $b^-(v) \ge k$ and $b^+(k) \ge k$ for all $v \in V$.

3.2 Initialization

In the subsequent steps, we assume that $b^-(v) \geq k$ and $b^+(v) \geq k$ for each $v \in V$, and that a perfect (b^-, b^+)-matching exists.

Let M be a minimum cost perfect (b^-, b^+)-matching. Notice that the proof of Theorem 5 indicates the reduction of the minimum cost perfect (b^-, b^+)-matching problem to the minimum cost perfect b-matching problem. Hence M is computable in polynomial time.

Let H be a directed Hamiltonian cycle constructed by Christofides' algorithm for the edge cost obtained from c by ignoring the direction of arcs (Recall that c is symmetric). Moreover let $H_1, \dots, H_k$ be k copies of H, $A = M \cup H_1 \cup \cdots \cup H_k$, and D_F denote the digraph (V, F) for an arc set F. Our algorithm for k-ACMDS prepares D_A as an initial graph.

A vertex $v \in V$ is called an *excess vertex* if $d^-(v) > b^-(v)$ or $d^+(v) > b^+(v)$ (otherwise v is called a *non-excess vertex*). Notice that $d^-(v; D_A) - b^-(v) = d^+(v; D_A) - b^+(v)$ because both sides are equal to k. This condition will be maintained throughout the algorithm, i.e., $d^-(v) > b^-(v)$ is equivalent to $d^+(v) > b^+(v)$.

3.3 Transformation into a feasible solution

This step decreases the degree of excess vertices in D_A as k-ECMDS. One difference between algorithms for k-ECMDS and for k-ACMDS is the definition of Operations 1 and 2. These will be executed for a pair of arcs entering and leaving the same vertex as follows.

Operation 1: If an excess vertex v has two incident arcs xv and vy in M with $x \neq y$, replace xv and vy by new edge $xy \in M$.

Operation 2: If an excess vertex v has two arcs uv and vu in M with $d^-(u) > b^-(u)$ (and $d^+(v) > b^+(v)$), remove these arcs.

Phase 1 of this step modifies edges in M by repeating Operations 1 and 2 until none of them is executable. We let M' denote M after Phase 1, and M denote the original set in the following. Moreover let $A' = M' \cup H_1 \cup \cdots \cup H_k$, and N denote the set of non-excess vertices in $D_{A'}$. Note that the number of arcs in M' entering (resp., leaving) each excess vertex v in $D_{A'}$ has $d^-(v; D_{A'}) - k \geq d^-(v; D_{A'}) - b^-(v)$ (resp., $d^+(v; D_{A'}) - k > d^+(v; D_{A'}) - b^+(v)$) arcs. Each excess vertex has only one neighbor in $G_{M'}$, and it is in N (i.e., a non-excess vertex in $D_{A'}$) since otherwise Operation 1 or 2 can be applied to v. This situation is simpler than after Phase 1 of the transformation into a feasible solution in UNDIRECT(k) since no correspondence of strict pairs exists. Notice that $N \neq \emptyset$ always holds here.

Phase 2 of this step modifies edges in $H_1, \dots, H_k$ so as to decrease the degrees of all excess vertices as in UNDIRECT(k). We repeat *detaching* each excess vertex v from $d^-(v; D_{A'}) - b^-(v; D_{A'})$ cycles in $H_1, \dots, H_k$ covering largest vertices, where detaching a vertex v from H_i is defined as an operation that replaces the pair $\{uv, vw\} \subseteq H_i$ of arcs entering and leaving v by new arc uw. Notice that this keeps $|V(H_i)| \leq |V(H_j)| + 1$ for any $i, j \in \{1, \dots, k\}$ as in UNDIRECT(k).

In the following, we let H'_i denote H_i after Phase 2 in order to avoid the ambiguity. Moreover let $A'' = M' \cup H'_1 \cup \cdots \cup H'_k$. Our algorithm outputs $D_{A''}$ as a solution.

3.4 Validity of our algorithm

The entire of our algorithm is described as follows.

Algorithm DIRECT(k)

Input: A vertex set V, in- and out-degree specification $b^-, b^+ : V \to \mathbb{Z}_+$, a symmetric metric arc cost $c : V \times V \to \mathbb{Q}_+$, and a positive integer k

Output: A k-arc-connected perfect (b^-, b^+)-matching or "INFEASIBLE"

\# Feasibility check

if $\sum_{v \in V} b^-(v) \neq \sum_{v \in V} b^+(v)$, $\exists v : b^-(v) > \sum_{u \in V-v} b^+(u)$, $\exists v : b^+(v) > \sum_{u \in V-v} b^-(u)$, $\exists v : k > b^-(v)$, or $\exists v : k > b^+(v)$ **then**

  Output "INFEASIBLE" and halt

end if;

\# Initialization

Compute a minimum cost perfect (b^-, b^+)-matching D_M;

Compute a Hamiltonian cycle D_H on V by Christofides' algorithm; Let $H_1, \ldots, H_k$ be k copies of H;

\# Phase 1 of transformation into a feasibile solution

$M' := M$;

while Operation 1 or 2 is applicable to a vertex $v \in V$ with $d^-(v; D_{M' \cup H_1 \cup \cdots \cup H_k}) > b^-(v)$ **do**

  if $\exists \{xv, vy\} \subseteq M'$ such that $x \neq y$ **then**

    $M' := (M' - \{xv, vy\}) \cup \{xy\}$ \# Operation 1

  else if $\exists \{xv, vx\} \subseteq M'$ such that $d^-(x; D_{M' \cup H_1 \cup \cdots \cup H_k}) > b^-(x)$ **then**

    $M' := M' - \{xv, vx\}$ \# Operation 2

  end if

end while;

\# Phase 2 of transformation into a feasible solution

$H'_i := H_i$ for each $i = 1, \ldots, k$;

for all $v \in V$ with $d^-(v; D_{M' \cup H'_1 \cup \cdots \cup H'_k}) > b^-(v)$ **do**

  while $d^-(v; D_{M' \cup H'_1 \cup \cdots \cup H'_k}) > b^-(v)$ **do**

    Choose $i \in \{1, \ldots, k\}$ such that $v \in V(H'_i)$ and $|V(H'_i)| \geq |V(H'_j)|$ for all $j \neq i$;

    Detach v from H'_i

  end while

end for;

$A'' := M' \cup H'_1 \cup \cdots \cup H'_k$;

Output $D_{A''}$

Let us see the feasibility of $D_{A''}$. We can show that $D_{A''}$ is k-arc-connected similarly for UNDIRECT(k). Here we see that the algorithm transforms D_A into a feasible solution without generating any loop.

Lemma 8 *Operations in Algorithm* DIRECT(k) *generates no loop.*

Proof: It suffices to show that Phase 2 keeps $V(H_i) \geq 2$ for $1 \leq i \leq k$. Recall that $N \neq \emptyset$. If $|N| \geq 2$, the claim is obvious since each of $H_1, \cdots, H_k$ is incident to all vertices in N. Hence suppose that $|N| = 1$, and let x be the unique vertex in N. Then all arcs in M' are incident to x since otherwise Operation 1 or 2 would be applicable to some vertex in $V - x$. In other words, it holds that $|M'| = d^-(x; D_{M'}) + d^+(v; D_{M'}) = b^-(x) + b^+(x) - 2k$. Recall that $\sum_{v \in V-x} b^+(v) \geq b^-(x)$ and $\sum_{v \in V-x} b^-(v) \geq b^+(x)$ hold by the assumption that a perfect (b^-, b^+)-matching exists.

Now assume that we have converted some excess vertices in $D_{A'}$ into non-excess vertices by detaching them from some of $H_1, \ldots, H_k$ while keeping $|V(H_i)| \geq 2$, $i = 1, \ldots, k$, and yet an excess vertex $y \in V - x$ remains. Let H_{i_1} be a cycle covering y (such H_{i_1} always exists). If $|V(H_{i_1})| \geq 3$, then we are done. Hence let us suppose the otherwise (i.e., $|V(H_{i_1})| = 2$).

The existence of y implies that $\sum_{v \in V} d^-(v) > \sum_{v \in V} b^-(v)$. Then there remains a cycles H_{i_2} with $|V(H_{i_2})| \geq 3$ because

$$\begin{aligned}\sum_{1 \leq i \leq k} |V(H_i)| &= \sum_{v \in V} d^-(v; D_{H_1 \cup \cdots \cup H_k}) = \sum_{v \in V} d^-(v; D_{E'}) - |M'| \\ &> \sum_{v \in V - \{x\}} b^-(v) + d^-(x; D_{E'}) - |M'| \geq b^+(x) + b^-(x) - |M'| = 2k.\end{aligned}$$

Let $y \notin V(H_{i_2})$, i.e., y has already detached from H_{i_2}. Before y is detached from H_{i_2}, it holds that $|V(H_{i_2})| \geq 4$. Hence $|V(H_{i_2})| > |V(H_{i_1})| + 1$ holds at this moment. However, the algorithm keeps $|V(H_i)| \leq |V(H_j)| + 1$ for any $i, j \in \{1, \ldots, k\}$ as stated above. Therefore, we have a contradiction. □

Let OPT denote the optimal cost of k-ACMDS. We can show that $c(M) \leq \mathrm{OPT}$ and $c(H_i) \leq 1.5\mathrm{OPT}/k$ for $1 \leq i \leq k$ similarly for UNDIRECT(k). As a conclusion, we have the following theorem.

Theorem 6 *Algorithm* DIRECT(k) *is a* 2.5*-approximation algorithm for* k-ACMDS. □

Algorithm DIRECT(k) always outputs a solution when there exists a perfect (b^-, b^+)-matching and $b^-(v) \geq k$, $b^+(v) \geq k$ for all $v \in V$. This fact and Theorem 5 implies the following corollary.

Corollary 2 *For* $k \geq 1$*, there exists a* k*-arc-connected perfect* (b^-, b^+)*-matching if and only if* $\sum_{v \in V} b^-(v) = \sum_{v \in V} b^+(v)$*,* $k \leq b^-(v) \leq \sum_{u \in V-v} b^+(u)$ *for each* $v \in V$*, and* $k \leq b^+(v) \leq \sum_{u \in V-v} b^-(u)$ *for each* $v \in V$*.* □

4 Uniform degree specification

In this section, we show that the approximation factor of our algorithms can be improved when $b(v) = \ell$ in k-ECMDS or $b^-(v) = b^+(v) = \ell$ in k-ACMDS for all $v \in V$ with some integer $\ell \geq k$.

We call a perfect b-matching (resp., a perfect (b^-, b^+)-matching) M *ℓ-regular* if $b(v) = \ell$ (resp., $b^-(v) = b^+(v) = \ell$) for all $v \in V$.

Lemma 9 *Assume that $b^-(v) = b^+(v) = \ell$ for all $v \in V$ and a (b^-, b^+)-matching exists. Let OPT denote the optimal cost of k-ACMDS. Then there exists an $(\ell - m)$-regular digraph D_R with $c(R) \le \frac{\ell - m}{\ell} OPT$ for an arbitrary non-negative integer $m \le \ell$.*

Proof: Let A denote an optimal arc set of k-ACMDS. As seen in Section 3, digraph D_A corresponds to the bipartite undirected graph (V^-, V^+, E), which is ℓ-regular. By Theorem 4, we can show that every ℓ-regular bipartite graph has a 1-regular subgraph. After removing the subgraph, the ℓ-regular bipartite graph becomes $\ell - 1$-regular. By applying this repeatedly, we can see that every ℓ-regular bipartite graph can be decomposed into ℓ graphs each of which is 1-regular [14]. Let R be the set of arcs corresponding to edges in least cost $\ell - m$ graphs of them. Then R is $(\ell - m)$-regular and $c(R) \le \frac{\ell - m}{\ell} c(A)$, as required. □

The union of an $(\ell - k)$-regular digraph and k Hamiltonian cycles are obviously feasible to k-ACMDS if $b^-(v) = b^+(v) = \ell$, $v \in V$. Therefore we can derive the following theorem.

Theorem 7 *If $b^-(v) = b^+(v) = \ell$ for all $v \in V$, then k-ACMDS is approximable within a factor of $1.5 + \frac{\ell - k}{\ell}$.* □

Next, we consider k-ECMDS.

Lemma 10 *Assume that $b(v) = \ell$ for all $v \in V$ and an ℓ-regular graph exists. Let OPT denote the optimal cost of k-ECMDS. Then there exists an $(\ell - 2m)$-regular graph G_R such that $c(R) \le \frac{\ell - 2m}{\ell} OPT$ if ℓ is even, and $c(R) \le (\frac{\ell - 2m - 1}{\ell} + \frac{1}{k}) OPT$ if ℓ is odd for an arbitrary non-negative integer m with $2m \le \ell$.*

Proof: Let E denote an optimal edge set of k-ECMDS. First suppose that ℓ is even. Then E can be oriented into an arc set A such that D_A is $\ell/2$-regular by traversing an Eulerian walk of E. Let c' be an arc cost function on A naturally defined from c (i.e., $c'(a) = c(e)$ if $a \in A$ corresponds to $e \in E$). As in the proof of Lemma 9, we can obtain an $(\ell/2 - m)$-regular digraph R' with $c'(R') \le \frac{\ell/2 - m}{\ell/2} c'(A)$. Let R be an edge set corresponding to R'. Then clearly G_R is $(\ell - 2m)$-regular and $c(R) \le \frac{\ell/2 - m}{\ell/2} c(E)$, as required.

Next, suppose that ℓ is odd. Let $2E$ denote the edge set obtained by duplicating each edge in E. Then G_{2E} is 2ℓ-regular. By the above argument about the case of ℓ is even, we can obtain an $(\ell - 2m - 1)$-regular graph G_F such that $c(F) \le \frac{\ell - 2m - 1}{2\ell} c(2E) = \frac{\ell - 2m - 1}{\ell} c(E)$ (Notice that $\ell - 2m - 1$ is even). Let M be a minimum cost 1-regular graph. Notice that such M exists since $|V|$ is even by the existence of an ℓ-regular graph with odd ℓ. Since the minimum cost of Hamiltonian cycles spanning all vertices is at most $2c(E)/k$ as shown in the proof of Lemma 7, we can see that $c(M) \le c(E)/k$. Let $R = F \cup M$. Then G_R is $(\ell - 2m)$-regular and $c(R) = c(F) + c(M) \le (\frac{\ell - 2m - 1}{\ell} + \frac{1}{k}) c(E)$, as required. □

Let $k' = \lceil k/2 \rceil$. The union of an $(\ell - 2k')$-regular graph and $2k'$ Hamiltonian cycles are obviously feasible to k-ECMDS if $b(v) = \ell$, $v \in V$. Therefore we can derive the following theorem.

Theorem 8 *If $b(v) = \ell$ for all $v \in V$, then k-ECMDS is approximable within a factor of $\frac{\ell - 2k'}{\ell} + 3\frac{k'}{k}$ if ℓ is even, and $\frac{(\ell - 2k' - 1)}{\ell} + \frac{1 + 3k'}{k}$ if ℓ is odd, where $k' = \lceil k/2 \rceil$.* □

Recall that metric TSP can be formulated as k-ECMDS with $b(v) = 2$, $v \in V$ and $k = 2$. Theorem 8 indicates that this case can be approximated within 1.5 as Christofides' algorithm.

5 Application for (m, n)-VRP

In this section, we consider the problem (m, n)-VRP. The formal definition of this problem is as follows. An instance of (m, n)-VRP consists of a vertex set V containing a special vertex s, a metric edge cost $c : \binom{V}{2} \to \mathbb{Q}_+$, and two non-negative integers m and n. The objective is to find a minimum cost set of m cycles, each containing s, such that each vertex in $V - s$ is contained in exactly n of those cycles. We can assume without loss of generality that $n \leq m \leq n(|V| - 1)$ since otherwise the instance is clearly infeasible.

An example of applying the (m, n)-VRP is the schedule of garbage collection. Let us consider the case in which a garbage collecting truck must visit each city on n of 5 weekdays in a week. A solution of $(5, n)$-VRP gives a schedule of this truck minimizing total length of routes.

Each solution to (m, n)-VRP is obviously feasible to $2n$-ECMDS with $b(s) = 2m$ and $b(v) = 2n$ for $v \in V - s$ (Hence the optimal value of $2n$-ECMDS with such b is at most that of (m, n)-VRP). However, the opposite direction does not hold as an example in Figure 5 shows. Nevertheless we can see that algorithm UNDIRECT$(2n)$ outputs a feasible solution for (m, n)-VRP.

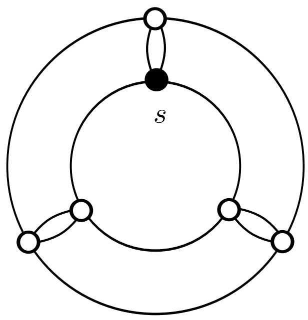

Figure 4: A solution to 4-ECMDS with $b(v) = 4$, $v \in V$, that is not feasible to $(2, 2)$-VRP

Theorem 9 *Let $b(s) = 2m$, $b(v) = 2n$ for each $v \in V - s$ and $k = 2n$. Then algorithm* UNDIRECT(k) *outputs a* 2.5*-approximate solution to* (m, n)*-VRP.*

Proof: The solution given by algorithm UNDIRECT(k) consists of edge set M' and cycles $H'_1, \ldots, H'_n$. In what follows, we see that this solution is feasible to (m, n)-VRP.

Let us consider the moment after Phase 1 of transformation into a feasible solution, and define E', M' and $H'_1, \ldots, H'_{k'}$ as in Section 2. Since $k = 2n$ is even, there exists no strict pair. Hence at least one end vertex of each edge in M' is a non-excess vertex. Let v be such a vertex. Then $b(v) = d(v; G_{E'}) > d(v; G_{H_1 \cup \cdots \cup H_n}) = 2n$ (Recall that each non-excess vertex is covered by all of $H_1, \ldots, H_n$). However, a vertex of degree more than $2n$ is only s since $b(u) = 2n$ for each $u \in V - s$. Hence we can see that (i) s is a non-excess vertex after Phase 1, and (ii) one end vertex of each in M' is s. Condition (i) implies that each of $H'_1, \ldots, H'_n$ covers s. Condition (ii) indicates that edges between s and a vertex $v \in V - s$ forms $d(v; M')/2$ cycles whose vertex sets are $\{s, v\}$ because $d(v; M')$ is even. Therefore, combining the fact that $d(v; G_{M' \cup H'_1 \cup \cdots \cup H'_n}) = b(v)$ for all $v \in V$, these show that UNDIRECT(k) outputs a feasible solution to (m, n)-VRP. □

The approximation factor can be improved as follows.

Theorem 10 *Problem (m, n)-VRP can be approximated within a factor of* $1.5 + \frac{m-n}{m}$.

Proof: Let $b(s) = 2m$, $b(v) = 2n$ for each $v \in V - s$ and $k = 2n$. Moreover, let E be an optimal solution for (m, n)-VRP, and F be the set of edges contained by $m - n$ cycles in G_E of least cost. Then it holds that $d(s; G_F) = 2m - 2n$ and $d(v; G_F) \leq 2n$ for $v \in V - s$. Besides this, we have $c(F) \leq \frac{m-n}{m} c(E)$ by the definition of F.

Now we let $V - s = \{v_1, \ldots, v_{|V|-1}\}$ so that $c(sv_1) \leq c(sv_2) \leq \cdots \leq c(sv_{|V|-1})$. Moreover we define R as an edge set which consists of $2n$ edges sv_i for each $i = 1, \ldots, p$ and $2m - 2n(p+1)$ edges sv_{p+1}, where $p = \lfloor (m-n)/n \rfloor$. Then it is clear that R is a minimum cost edge set such that $d(s; G_R) = 2np + 2m - 2n(p+1) = 2m - 2n$ and $d(v; G_R) \leq 2n$ for all $v \in V - s$. This implies that $c(R) \leq c(F) \leq \frac{m-n}{m} c(E)$.

By using R instead of M in UNDIRECT(k), we can obtain a feasible solution to k-ECMDS. As in Theorem 9, this solution is also feasible to (m, n)-VRP. Moreover the cost of the solution is at most $c(H_1) + \cdots + c(H_{k'}) + c(R) \leq (1.5 + \frac{m-n}{m}) c(E)$, which completes the proof. □

6 Concluding Remarks

We note that some cases of k-ECMDS/k-ACMDS remain open. One is 1-ECMDS with $b(v) = 1$ for some $v \in V$. Our algorithm cannot deal with this case because detaching the vertices in a strict pair from the same Hamiltonian cycle in Phase 2 may lose the connectivity. Also a key problem for approximating 1-ECMDS would be to find a minimum cost spanning tree such that $d(v) \leq b(v)$, $v \in V$ for a given $b : V \to \mathbb{Z}_+$. However, no constant factor approximation algorithm is known to this problem if $b(v) = 1$ for some $v \in V$, although it can be approximated within a constant factor of 2 if $b(v) \geq 2$ for all $v \in V$ [2]. In addition to this, it has been shown in [15] that a spanning tree T of optimal cost is computable in polynomial time while they allow to violate the degree upper bound by at most 1.

Another interesting open problem is a generalization of k-ECMDS (resp., k-ACMDS) in which the k-edge-connectivity (resp., k-arc-connectivity) requirement is replaced by a local-edge-connectivity requirement. It is also interesting to consider the problem in which the number of multiple edges are constrained, to which our algorithm can not be applied.

It is also valuable to characterize the feasible solutions to (m, n)-VRP. In Section 5, we noted that specifying the edge-connectivity and the degree of each vertex is not enough for this although our algorithm always outputs a feasible solution to (m, n)-VRP. Moreover, it is interesting to study a further generalization of (m, n)-VRP in which the number $b(v)/2$ of cycles containing each vertex v is not uniform.

Acknowledgement

This research was partially supported by the Scientific Grant-in-Aid from Ministry of Education, Culture, Sports, Science and Technology of Japan. The authors would like to thank anonymous referees for their helpful comments.

References

[1] R. P. Anstee, A polynomial algorithm for b-matchings: an alternative approach, Information Processing Letters 24 (1987) 153–157.

[2] S. P. Fekete, S. Khuller, M. Klemmstein, B. Raghavachari, N. Young, A network-flow technique for finding low-weight bounded-degree spanning trees, Journal of Algorithms 24 (1997) 310–324.

[3] A. Frank, Augmenting graphs to meet edge-connectivity requirements, SIAM Journal on Discrete Mathematics 5 (1992) 25–53.

[4] A. Frank, On a theorem of Mader, Discrete Mathematics 191 (1992) 49–57.

[5] G. N. Frederickson, M. S. Hecht, C. E. Kim, Approximation algorithms for some routing problems, SIAM Journal of Computing 7 (1978) 178–193.

[6] T. Fukunaga, H. Nagamochi, Approximating minimum cost multigraphs of specified edge-connectivity under degree bounds, Proceedings of the 9th Japan-Korea Joint Workshop on Algorithm and Computation (2006) 25-32.

[7] T. Fukunaga, H. Nagamochi, Approximating a generalization of metric TSP, IEICE Transactions on Information and Systems E90-D (2007) 432–439.

[8] M. X. Goemans, D. J. Bertsimas, Survivable networks, linear programming relaxations and the parsimonious property, Mathematical Programming 60 (1993) 145–166.

[9] M. X. Goemans, D. P. Williamson, The primal-dual method for approximation algorithms and its application to network design problems, PWS, 1997, Ch. 4, pp. 144–191.

[10] K. Jain, A factor 2 approximation algorithm for the generalized Steiner network problem, Combinatorica 21 (2001) 39–60.

[11] L. C. Lau, S. Naor, M. Salavatipour, M. Singh, Survivable network design with degree or order constraints, Proceedings of 39th ACM Symposium on Theory of Computing (2007) 651–660.

[12] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, D. B. Shmoys (Eds.), The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization, John Wiley & Sons, 1985.

[13] W. Mader, A reduction method for edge-connectivity in graphs, Annals of Discrete Mathematics 3 (1978) 145–164.

[14] A. Schrijver, Combinatorial Optimization: Polyhedra and Efficiency, Springer, 2003.

[15] M. Singh, L. C. Lau, Approximating minimum bounded degree spanning trees to within one of optimal, Proceedings of 39th ACM Symposium on Theory of Computing (2007) 661–670.

[16] V. Vazirani, Approximation Algorithm, Springer, 2001.

[17] L. A. Wolsey, Heuristic analysis, linear programming and branch and bound, Mathematical Programming Study 13 (1980) 121–134.