Approximation Algorithms for Mixed, Windy, and Capacitated Arc Routing Problems

René van Bevern[1]

joint work with

Christian Komusiewicz[2] and Manuel Sorge[2]

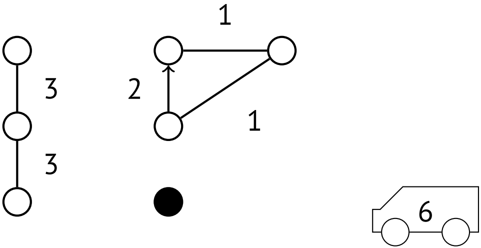
[1]Novosibirsk State University, Novosibirsk, Russian Federation

[2]Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Germany

ALGO / ATMOS 2015

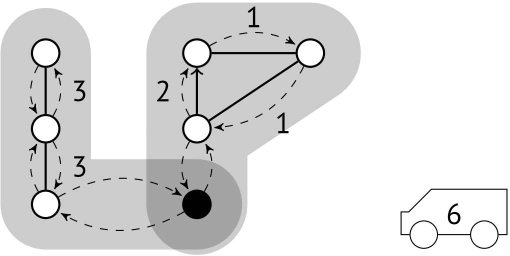
Mixed and Windy Capacitated Arc Routing (MWCARP)

Input: Mixed graph $G = (V, A, E)$, depot $v_0 \in V$, travel costs $c\colon V \times V \to \mathbb{N} \cup \{\infty\}$, demands $d\colon E \cup A \to \mathbb{N}$, and a vehicle capacity Q.



Mixed and Windy Capacitated Arc Routing (MWCARP)

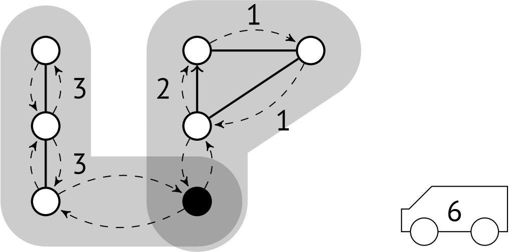
Input: Mixed graph $G = (V, A, E)$, depot $v_0 \in V$, travel costs $c\colon V \times V \to \mathbb{N} \cup \{\infty\}$, demands $d\colon E \cup A \to \mathbb{N}$, and a vehicle capacity Q.



Goal: Find a set of vehicle routes of total minimum cost, each starting and ending in v_0, such that each demand is served by exactly one vehicle and each vehicle serves a total demand of at most Q.

Mixed and Windy Capacitated Arc Routing (MWCARP)

Input: Mixed graph $G = (V, A, E)$, depot $v_0 \in V$, travel costs $c\colon V \times V \to \mathbb{N} \cup \{\infty\}$, demands $d\colon E \cup A \to \mathbb{N}$, and a vehicle capacity Q.



Goal: Find a set of vehicle routes of total minimum cost, each starting and ending in v_0, such that each demand is served by exactly one vehicle and each vehicle serves a total demand of at most Q.

Applications: Waste collection, salting roads, mail delivery, meter reading, inspection of welded seams, … *(Corberán and Laporte, SIAM book, 2014)*

Known results

3/2-approximation is NP-hard *(Golden and Wong, Networks, 1981)*

Known results

3/2-approximation is NP-hard *(Golden and Wong, Networks, 1981)*

Undirected version is 7/2-approximable *(Jansen, Networks, 1993)*

Known results

3/2-approximation is NP-hard *(Golden and Wong, Networks, 1981)*

Undirected version is 7/2-approximable *(Jansen, Networks, 1993)*

In general, the approximation factor for MWCARP cannot undercut that for metric asymmetric TSP ($\triangle$-ATSP).

Known results

3/2-approximation is NP-hard *(Golden and Wong, Networks, 1981)*

Undirected version is 7/2-approximable *(Jansen, Networks, 1993)*

In general, the approximation factor for MWCARP cannot undercut that for metric asymmetric TSP ($\triangle$-ATSP).

Approximation algorithms for n-vertex $\triangle$-ATSP:

Known results

3/2-approximation is NP-hard *(Golden and Wong, Networks, 1981)*

Undirected version is 7/2-approximable *(Jansen, Networks, 1993)*

In general, the approximation factor for MWCARP cannot undercut that for metric asymmetric TSP ($\triangle$-ATSP).

Approximation algorithms for n-vertex $\triangle$-ATSP:

- Best known: $O(\log n/\log\log n)$-approximation *(Asadpour et al., SODA'10)*

Known results

3/2-approximation is NP-hard *(Golden and Wong, Networks, 1981)*

Undirected version is 7/2-approximable *(Jansen, Networks, 1993)*

In general, the approximation factor for MWCARP cannot undercut that for metric asymmetric TSP ($\triangle$-ATSP).

Approximation algorithms for n-vertex $\triangle$-ATSP:

- Best known: $O(\log n / \log\log n)$-approximation *(Asadpour et al., SODA'10)*
- Constant-factor approximation open *(Frieze, Galbiati, Maffioli, Networks, 1982)*

Known results

3/2-approximation is NP-hard *(Golden and Wong, Networks, 1981)*

Undirected version is 7/2-approximable *(Jansen, Networks, 1993)*

In general, the approximation factor for MWCARP cannot undercut that for metric asymmetric TSP ($\triangle$-ATSP).

Approximation algorithms for n-vertex $\triangle$-ATSP:

- Best known: $O(\log n / \log \log n)$-approximation *(Asadpour et al., SODA'10)*
- Constant-factor approximation open *(Frieze, Galbiati, Maffioli, Networks, 1982)*

$\rightsquigarrow$ Approximation algorithms for MWCARP sparsely investigated.

How to approximate despite the hardness?

Consider parameter *C*: number of (weakly) connected components induced by positive-demand arcs.

How to approximate despite the hardness?

Consider parameter C: number of (weakly) connected components induced by positive-demand arcs.

In picture: $1.5 \cdot 10^6$ inhabitants, lots of roads to serve, but $C \approx 5$.

How to approximate despite the hardness?

Consider parameter C: number of (weakly) connected components induced by positive-demand arcs.

In picture: $1.5 \cdot 10^6$ inhabitants, lots of roads to serve, but $C \approx 5$.

NP-hard for $C = 1$.

(Golden and Wong, Networks, 1981)

How to approximate despite the hardness?

Consider parameter C: number of (weakly) connected components induced by positive-demand arcs.

In picture: $1.5 \cdot 10^6$ inhabitants, lots of roads to serve, but $C \approx 5$.

NP-hard for $C = 1$.

(Golden and Wong, Networks, 1981)

Our result:

MWCARP is $O(1)$-approximable in $O(2^C C^2 + n^3 \log n)$ time.

How to approximate despite the hardness?

Consider parameter C: number of (weakly) connected components induced by positive-demand arcs.

In picture: $1.5 \cdot 10^6$ inhabitants, lots of roads to serve, but $C \approx 5$.

NP-hard for $C = 1$.

(Golden and Wong, Networks, 1981)

Our result:

MWCARP is $O(1)$-approximable in $O(2^C C^2 + n^3 \log n)$ time.

$\rightsquigarrow$ poly-time for $C \in O(\log n)$.

Our results (in more detail)

We show that MWCARP and the following related **NP-hard problems** are not only as **hard** to approximate as △-ATSP, but also as **easy**.

Our results (in more detail)

We show that MWCARP and the following related **NP-hard problems** are not only as **hard** to approximate as △-ATSP, but also as **easy**.

Mixed and Windy Rural Postman problem (MWRPP): find a minimum-cost tour visiting all given **required arcs and edges** in a mixed asymmetrically edge-weighted graph.

Our results (in more detail)

We show that MWCARP and the following related **NP-hard problems** are not only as **hard** to approximate as △-ATSP, but also as **easy**.

Mixed and Windy Rural Postman problem (MWRPP): find a minimum-cost tour visiting all given **required arcs and edges** in a mixed asymmetrically edge-weighted graph.

Directed Rural Postman problem (DRPP): find a minimum-cost tour visiting all given **required arcs** in a directed edge-weighted graph.

Our results (in more detail)

We show that MWCARP and the following related **NP-hard problems** are not only as **hard** to approximate as $\triangle$-ATSP, but also as **easy**.

Mixed and Windy Rural Postman problem (MWRPP): find a minimum-cost tour visiting all given **required arcs and edges** in a mixed asymmetrically edge-weighted graph.

Directed Rural Postman problem (DRPP): find a minimum-cost tour visiting all given **required arcs** in a directed edge-weighted graph.

Theorem: If n-vertex $\triangle$-ATSP is $\alpha(n)$-approximable in $t(n)$ time, then

1. n-vertex DRPP is $(\alpha(C) + 1)$-approximable in $O(t(C) + n^3 \log n)$ time,

Our results (in more detail)

We show that MWCARP and the following related **NP-hard problems** are not only as **hard** to approximate as $\triangle$-ATSP, but also as **easy**.

Mixed and Windy Rural Postman problem (MWRPP): find a minimum-cost tour visiting all given **required arcs and edges** in a mixed asymmetrically edge-weighted graph.

Directed Rural Postman problem (DRPP): find a minimum-cost tour visiting all given **required arcs** in a directed edge-weighted graph.

Theorem: If n-vertex $\triangle$-ATSP is $\alpha(n)$-approximable in $t(n)$ time, then

1. n-vertex DRPP is $(\alpha(C) + 1)$-approximable in $O(t(C) + n^3 \log n)$ time,
2. n-vertex MWRPP is $(\alpha(C) + 3)$-appr'mable in $O(t(C) + n^3 \log n)$ time,

Our results (in more detail)

We show that MWCARP and the following related **NP-hard problems** are not only as **hard** to approximate as $\triangle$-ATSP, but also as **easy**.

Mixed and Windy Rural Postman problem (MWRPP): find a minimum-cost tour visiting all given **required arcs and edges** in a mixed asymmetrically edge-weighted graph.

Directed Rural Postman problem (DRPP): find a minimum-cost tour visiting all given **required arcs** in a directed edge-weighted graph.

Theorem: If n-vertex $\triangle$-ATSP is $\alpha(n)$-approximable in $t(n)$ time, then

1. n-vertex DRPP is $(\alpha(C) + 1)$-approximable in $O(t(C) + n^3 \log n)$ time,
2. n-vertex MWRPP is $(\alpha(C) + 3)$-appr'mable in $O(t(C) + n^3 \log n)$ time,
3. n-vertex MWCARP is $O(\alpha(C + 1))$-ap'le in $O(t(C + 1) + n^3 \log n)$ time.

Our results (in more detail)

We show that MWCARP and the following related **NP-hard problems** are not only as **hard** to approximate as $\triangle$-ATSP, but also as **easy**.

Mixed and Windy Rural Postman problem (MWRPP): find a minimum-cost tour visiting all given **required arcs and edges** in a mixed asymmetrically edge-weighted graph.

Directed Rural Postman problem (DRPP): find a minimum-cost tour visiting all given **required arcs** in a directed edge-weighted graph.

Theorem: If n-vertex $\triangle$-ATSP is $\alpha(n)$-approximable in $t(n)$ time, then

1. n-vertex DRPP is $(\alpha(C) + 1)$-approximable in $O(t(C) + n^3 \log n)$ time,
2. n-vertex MWRPP is $(\alpha(C) + 3)$-appr'mable in $O(t(C) + n^3 \log n)$ time,
3. n-vertex MWCARP is $O(\alpha(C + 1))$-ap'le in $O(t(C + 1) + n^3 \log n)$ time.

Remark: $\triangle$-ATSP is optimally solvable in $O(2^n n^2)$ time. *(Bell, Held, Karp, 1962)*

Component-Wise Eulerian DRPP

Goal: find a minimum-cost tour visiting a **set *R* of required arcs** in a directed edge-weighted **graph *G***.

Component-Wise Eulerian DRPP

Goal: find a minimum-cost tour visiting a **set R of required arcs** in a directed edge-weighted **graph G**.

Additional constraint: $G[R]$ has C weakly-connected Eulerian components.

Component-Wise Eulerian DRPP

Goal: find a minimum-cost tour visiting a **set R of required arcs** in a directed edge-weighted **graph G**.

Additional constraint: $G[R]$ has C weakly-connected Eulerian components.

Claim: $\alpha(n)$-approximation for n-vertex $\triangle$-ATSP
$\implies$ $(\alpha(C)+1)$-approximation for Component-Wise Eulerian DRPP.

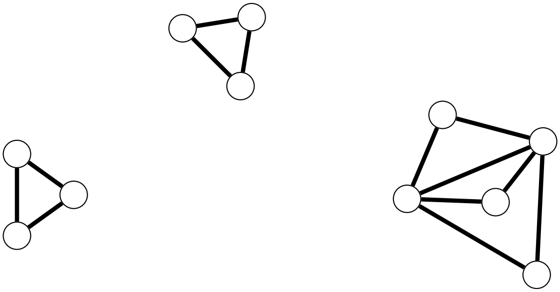
Algorithm for Component-Wise Eulerian DRPP

Claim: $\alpha(n)$-approximation for n-vertex $\triangle$-ATSP
$\implies$ $(\alpha(C) + 1)$-approximation for Component-Wise Eulerian DRPP.

Algorithm for Component-Wise Eulerian DRPP

Claim: $\alpha(n)$-approximation for n-vertex $\triangle$-ATSP
$\implies$ $(\alpha(C)+1)$-approximation for Component-Wise Eulerian DRPP.

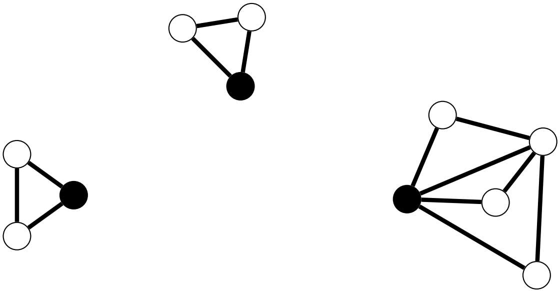
Algorithm: Shown are only **required arcs**; directions are **not** shown.



Algorithm for Component-Wise Eulerian DRPP

Claim: $\alpha(n)$-approximation for n-vertex $\triangle$-ATSP
$\implies$ $(\alpha(C)+1)$-approximation for Component-Wise Eulerian DRPP.

Algorithm: Shown are only **required arcs**; directions are **not** shown.

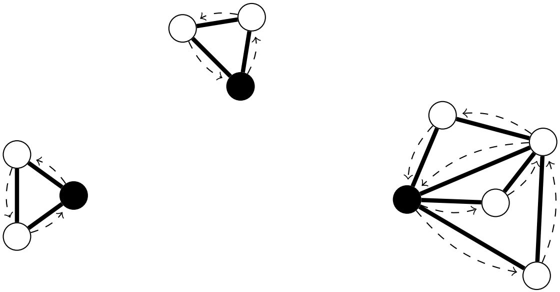


For each connected component i of $G[R]$, pick an arbitrary vertex v_i.

Algorithm for Component-Wise Eulerian DRPP

Claim: $\alpha(n)$-approximation for n-vertex $\triangle$-ATSP
$\implies$ $(\alpha(C)+1)$-approximation for Component-Wise Eulerian DRPP.

Algorithm: Shown are only **required arcs**; directions are **not** shown.

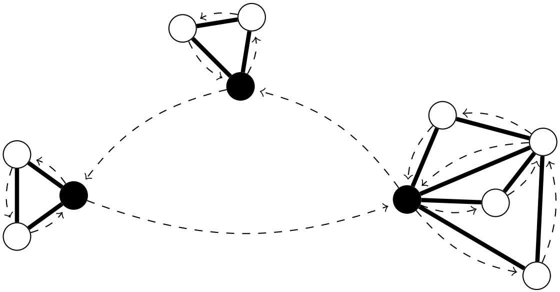


For each component i, compute an Euler tour starting and ending in v_i.

Algorithm for Component-Wise Eulerian DRPP

Claim: $\alpha(n)$-approximation for n-vertex $\triangle$-ATSP
$\implies$ $(\alpha(C) + 1)$-approximation for Component-Wise Eulerian DRPP.

Algorithm: Shown are only **required arcs**; directions are **not** shown.

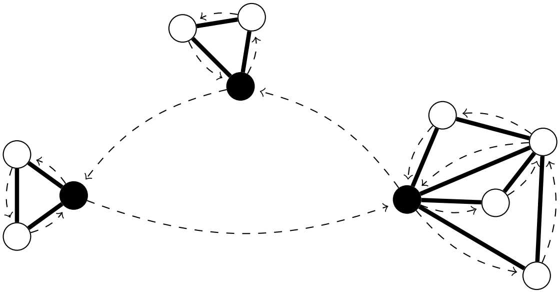


Join vertices $v_1, \ldots, v_C$ using $\alpha(C)$-approximate $\triangle$-ATSP tour.

Algorithm for Component-Wise Eulerian DRPP

Claim: $\alpha(n)$-approximation for n-vertex $\triangle$-ATSP
$\implies (\alpha(C) + 1)$-approximation for Component-Wise Eulerian DRPP.

Algorithm: Shown are only **required arcs**; directions are **not** shown.



Join vertices $v_1, \ldots, v_C$ using $\alpha(C)$-approximate $\triangle$-ATSP tour.

Approximation factor: Optimum DRPP tour has at least the cost of the required arcs and at least the cost of an optimal $\triangle$-ATSP tour.

DRPP

Goal: find a minimum-cost tour visiting all given **required arcs** in a directed edge-weighted graph.

DRPP

Goal: find a minimum-cost tour visiting all given **required arcs** in a directed edge-weighted graph.

Reminder: $G[R]$ consists of C weakly-connected components.

DRPP

Goal: find a minimum-cost tour visiting all given **required arcs** in a directed edge-weighted graph.

Reminder: $G[R]$ consists of C weakly-connected components.

Claim: $\alpha(n)$-approx. $\triangle$-ATSP $\implies$ $(\alpha(C) + 1)$-approx. for DRPP.

Algorithm for general DRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C) + 1)$-approx. for DRPP.

Algorithm for general DRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C)+1)$-approx. for DRPP.

Observation: any **DRPP tour *T*** enters each vertex of the **input graph *G*** as often as it leaves.

Algorithm for general DRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C)+1)$-approx. for DRPP.

Observation: any **DRPP tour *T*** enters each vertex of the **input graph *G*** as often as it leaves.

We say that the **multigraph *G*[*T*]** induced by the arcs on T **is balanced**.

Algorithm for general DRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C)+1)$-approx. for DRPP.

Observation: any **DRPP tour *T*** enters each vertex of the **input graph *G*** as often as it leaves.

We say that the **multigraph *G*[*T*]** induced by the arcs on T **is balanced**.

$\rightsquigarrow$ If we add to the **set *R* of required arcs** a minimum-cost multiset of **arcs R^* such that $G[R \uplus R^*]$ is balanced**, then

$$c(R \uplus R^*) \leq c(T)$$

Algorithm for general DRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C)+1)$-approx. for DRPP.

Observation: any **DRPP tour T** enters each vertex of the **input graph G** as often as it leaves.

We say that the **multigraph $G[T]$** induced by the arcs on T **is balanced**.

$\rightsquigarrow$ If we add to the **set R of required arcs** a minimum-cost multiset of **arcs R^* such that $G[R \uplus R^*]$ is balanced**, then

$$c(R \uplus R^*) \leq c(T)$$

and **all connected components of $G[R \uplus R^*]$ are Eulerian.**

Algorithm for general DRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C)+1)$-approx. for DRPP.

Observation: any **DRPP tour T** enters each vertex of the **input graph G** as often as it leaves.

We say that the **multigraph $G[T]$** induced by the arcs on T **is balanced**.

$\rightsquigarrow$ If we add to the **set R of required arcs** a minimum-cost multiset of **arcs R^* such that $G[R \uplus R^*]$ is balanced**, then

$$c(R \uplus R^*) \leq c(T)$$

and **all connected components of $G[R \uplus R^*]$ are Eulerian.**

Algorithm: use the algorithm for Component-Wise Eulerian DRPP on G with the required arcs $R \uplus R^*$.

Algorithm for general DRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C)+1)$-approx. for DRPP.

Observation: any **DRPP tour T** enters each vertex of the **input graph G** as often as it leaves.

We say that the **multigraph $G[T]$** induced by the arcs on T **is balanced**.

$\rightsquigarrow$ If we add to the **set R of required arcs** a minimum-cost multiset of **arcs R^* such that $G[R \uplus R^*]$ is balanced**, then

$$c(R \uplus R^*) \leq c(T)$$

and **all connected components of $G[R \uplus R^*]$ are Eulerian.**

Algorithm: use the algorithm for Component-Wise Eulerian DRPP on G with the required arcs $R \uplus R^*$.

How to compute R^*?

Algorithm for general DRPP II

Goal: given a set R of required arcs in G, compute a minimum-cost multiset of arcs R^* such that $G[R \uplus R^*]$ is balanced.

Algorithm for general DRPP II

Goal: given a set R of required arcs in G, compute a minimum-cost multiset of arcs R^* such that $G[R \uplus R^*]$ is balanced.

Definition: Let $\text{balance}(v) := \text{indeg}_{G[R]}(v) - \text{outdeg}_{G[R]}(v)$.

Algorithm for general DRPP II

Goal: given a set R of required arcs in G, compute a minimum-cost multiset of arcs R^* such that $G[R \uplus R^*]$ is balanced.

Definition: Let $\text{balance}(v) := \text{indeg}_{G[R]}(v) - \text{outdeg}_{G[R]}(v)$.

For a vertex v with

- $\text{balance}(v) > 0$, the multiset R^* has to contain $\text{balance}(v)$ more out-arcs than in-arcs incident to v,

Algorithm for general DRPP II

Goal: given a set R of required arcs in G, compute a minimum-cost multiset of arcs R^* such that $G[R \uplus R^*]$ is balanced.

Definition: Let $\text{balance}(v) := \text{indeg}_{G[R]}(v) - \text{outdeg}_{G[R]}(v)$.

For a vertex v with

- $\text{balance}(v) > 0$, the multiset R^* has to contain $\text{balance}(v)$ more out-arcs than in-arcs incident to v,
- $\text{balance}(v) < 0$, the multiset R^* has to contain $\text{balance}(v)$ more in-arcs than out-arcs incident to v,

Algorithm for general DRPP II

Goal: given a set R of required arcs in G, compute a minimum-cost multiset of arcs R^* such that $G[R \uplus R^*]$ is balanced.

Definition: Let $\text{balance}(v) := \text{indeg}_{G[R]}(v) - \text{outdeg}_{G[R]}(v)$.

For a vertex v with

- $\text{balance}(v) > 0$, the multiset R^* has to contain $\text{balance}(v)$ more out-arcs than in-arcs incident to v,
- $\text{balance}(v) < 0$, the multiset R^* has to contain $\text{balance}(v)$ more in-arcs than out-arcs incident to v,
- $\text{balance}(v) = 0$, the multiset R^* has to contain as many out-arcs as in-arcs incident to v.

Algorithm for general DRPP II

Goal: given a set R of required arcs in G, compute a minimum-cost multiset of arcs R^* such that $G[R \uplus R^*]$ is balanced.

Definition: Let $\text{balance}(v) := \text{indeg}_{G[R]}(v) - \text{outdeg}_{G[R]}(v)$.

For a vertex v with

- $\text{balance}(v) > 0$, the multiset R^* has to contain $\text{balance}(v)$ more out-arcs than in-arcs incident to v,
- $\text{balance}(v) < 0$, the multiset R^* has to contain $\text{balance}(v)$ more in-arcs than out-arcs incident to v,
- $\text{balance}(v) = 0$, the multiset R^* has to contain as many out-arcs as in-arcs incident to v.

$\rightsquigarrow$ polynomial-time solvable **uncapacitated min-cost flow** problem, where **demand and supply** of vertex v are given by **balance(v)**.

MWRPP

Goal: find a minimum-cost tour visiting all given **required arcs and edges** in a mixed asymmetrically edge-weighted graph.

MWRPP

Goal: find a minimum-cost tour visiting all given **required arcs and edges** in a mixed asymmetrically edge-weighted graph.

Reminder: $G[R]$ consists of C weakly-connected components.

MWRPP

Goal: find a minimum-cost tour visiting all given **required arcs and edges** in a mixed asymmetrically edge-weighted graph.

Reminder: $G[R]$ consists of C weakly-connected components.

Claim: $\alpha(n)$-approx. $\triangle$-ATSP $\implies$ $(\alpha(C)+3)$-approx. for MWRPP.

Algorithm for MWRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C)+3)$-approx. for MWRPP.

Algorithm for MWRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C) + 3)$-approx. for MWRPP.

Trick: Just reduce MWRPP to the DRPP as follows:

- replace each undirected edges by a pair of arcs directed into opposite directions,

Algorithm for MWRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C) + 3)$-approx. for MWRPP.

Trick: Just reduce MWRPP to the DRPP as follows:

- replace each undirected edges by a pair of arcs directed into opposite directions,
- so that required edges are served in the cheaper direction.

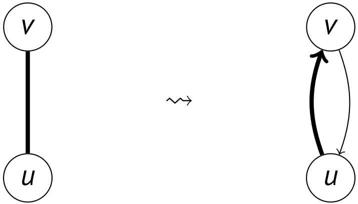
Algorithm for MWRPP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $(\alpha(C)+3)$-approx. for MWRPP.

Trick: Just reduce MWRPP to the DRPP as follows:

- replace each undirected edges by a pair of arcs directed into opposite directions,
- so that required edges are served in the cheaper direction.

If $c(u,v) \leq c(v,u)$, then (required arcs and edges are bold):



Every tour for the MWRPP instance can be turned into a tour for the DRPP instance that pays at most the triple price for each required edge.

MWCARP

Input: mixed graph $G = (V, A, E)$, depot $v_0 \in V$, travel costs $c \colon V \times V \to \mathbb{N} \cup \{\infty\}$, demands $d \colon E \cup A \to \mathbb{N}$, and a vehicle capacity Q.

Reminder: positive-demand arcs induce C weakly-connected components.

Goal: find a set of vehicle routes of total minimum cost, each starting and ending in v_0, such that each demand is served by exactly one vehicle and each vehicle serves a total demand of at most Q.

MWCARP

Input: mixed graph $G = (V, A, E)$, depot $v_0 \in V$, travel costs $c\colon V \times V \to \mathbb{N} \cup \{\infty\}$, demands $d\colon E \cup A \to \mathbb{N}$, and a vehicle capacity Q.

Reminder: positive-demand arcs induce C weakly-connected components.

Goal: find a set of vehicle routes of total minimum cost, each starting and ending in v_0, such that each demand is served by exactly one vehicle and each vehicle serves a total demand of at most Q.

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $O(\alpha(C+1))$-approx. for MWCARP.

Algorithm for MWCARP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $O(\alpha(C+1))$-approx. for MWCARP.

Algorithm for MWCARP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $O(\alpha(C+1))$-approx. for MWCARP.

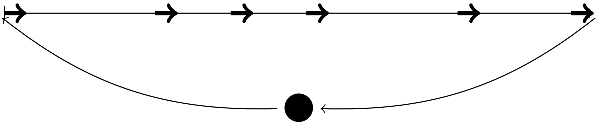
Observation: Concatenating vehicle tours of optimal MWCARP solution gives an MWRPP tour visiting all positive-demand arcs and depot v_0.

Algorithm for MWCARP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $O(\alpha(C+1))$-approx. for MWCARP.

Observation: Concatenating vehicle tours of optimal MWCARP solution gives an MWRPP tour visiting all positive-demand arcs and depot v_0.

1. Compute $(\alpha(C+1)+3)$-approximate MWRPP tour T visiting depot v_0 and all positive-demand arcs.

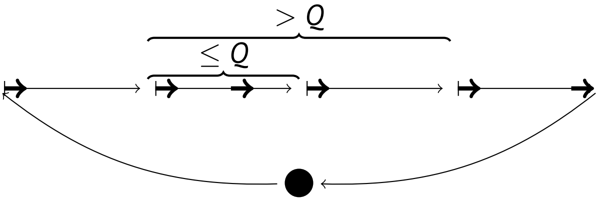


Algorithm for MWCARP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $O(\alpha(C+1))$-approx. for MWCARP.

Observation: Concatenating vehicle tours of optimal MWCARP solution gives an MWRPP tour visiting all positive-demand arcs and depot v_0.

1. Compute $(\alpha(C+1)+3)$-approximate MWRPP tour T visiting depot v_0 and all positive-demand arcs.
2. Split it greedily into maximal subwalks of demand at most Q each.

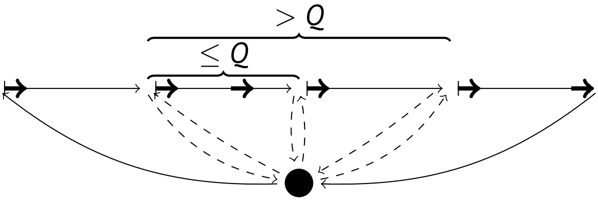


Algorithm for MWCARP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $O(\alpha(C+1))$-approx. for MWCARP.

Observation: Concatenating vehicle tours of optimal MWCARP solution gives an MWRPP tour visiting all positive-demand arcs and depot v_0.

1. Compute $(\alpha(C+1)+3)$-approximate MWRPP tour T visiting depot v_0 and all positive-demand arcs.
2. Split it greedily into maximal subwalks of demand at most Q each.
3. Add a shortest path from and to v_0 to each subwalk.

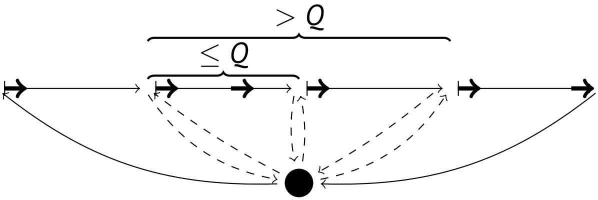


Algorithm for MWCARP

Claim: $\alpha(n)$-approx. for $\triangle$-ATSP $\implies$ $O(\alpha(C+1))$-approx. for MWCARP.

Observation: Concatenating vehicle tours of optimal MWCARP solution gives an MWRPP tour visiting all positive-demand arcs and depot v_0.

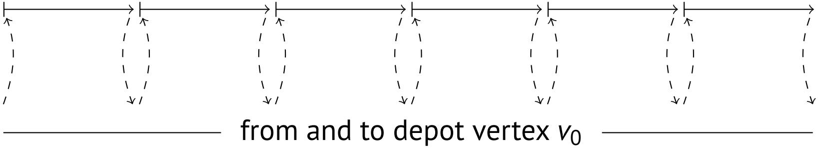
1. Compute $(\alpha(C+1)+3)$-approximate MWRPP tour T visiting depot v_0 and all positive-demand arcs.
2. Split it greedily into maximal subwalks of demand at most Q each.
3. Add a shortest path from and to v_0 to each subwalk.



Approximation factor: Since $c(T) \in O(\alpha(C+1) \cdot \text{OPT})$, it remains to analyze the total length of the shortest paths added to the subwalks.

Approximation factor analysis for MWCARP

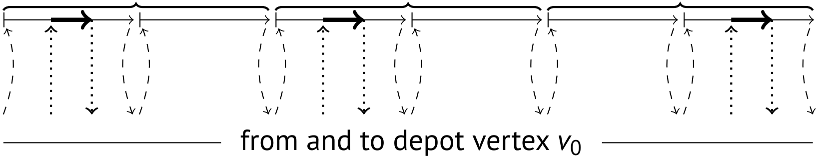
Goal: Analyze total length of the **shortest paths (dashed)** added to the subwalks of the **MWRPP tour (solid)**.



Approximation factor analysis for MWCARP

Goal: Analyze total length of the **shortest paths (dashed)** added to the subwalks of the **MWRPP tour (solid)**.

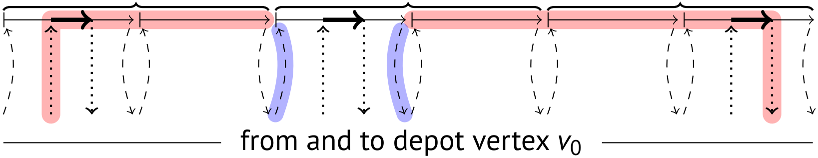
One can show: each consecutive pair of subwalks contains an arc served by a **distinct** vehicle tour of an **optimal solution (dotted).**



Approximation factor analysis for MWCARP

Goal: Analyze total length of the **shortest paths (dashed)** added to the subwalks of the **MWRPP tour (solid)**.

One can show: each consecutive pair of subwalks contains an arc served by a **distinct** vehicle tour of an **optimal solution (dotted).**

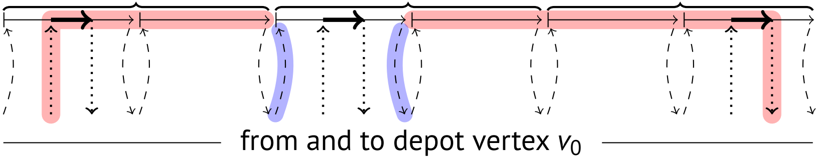


Charge length of the blue shortest paths to the red walks, consisting of arcs of the optimal solution and of arcs of the approximate MWRPP tour.

Approximation factor analysis for MWCARP

Goal: Analyze total length of the **shortest paths (dashed)** added to the subwalks of the **MWRPP tour (solid)**.

One can show: each consecutive pair of subwalks contains an arc served by a **distinct** vehicle tour of an **optimal solution (dotted).**



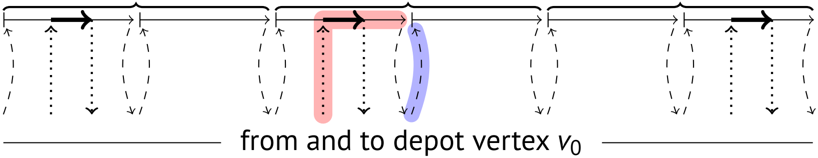
Charge length of the blue shortest paths to the red walks, consisting of arcs of the optimal solution and of arcs of the approximate MWRPP tour.

Each arc is **charged** to at most **five times** $\rightsquigarrow$ total cost of shortest paths is at most $5(\alpha(C+1)+3) \cdot \text{OPT} + 5 \cdot \text{OPT}$.

Approximation factor analysis for MWCARP

Goal: Analyze total length of the **shortest paths (dashed)** added to the subwalks of the **MWRPP tour (solid)**.

One can show: each consecutive pair of subwalks contains an arc served by a **distinct** vehicle tour of an **optimal solution (dotted).**



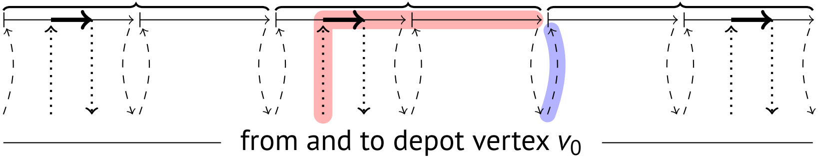
Charge length of the blue shortest paths to the red walks, consisting of arcs of the optimal solution and of arcs of the approximate MWRPP tour.

Each arc is **charged** to at most **five times** $\rightsquigarrow$ total cost of shortest paths is at most $5(\alpha(C+1)+3) \cdot \mathrm{OPT} + 5 \cdot \mathrm{OPT}$.

Approximation factor analysis for MWCARP

Goal: Analyze total length of the **shortest paths (dashed)** added to the subwalks of the **MWRPP tour (solid)**.

One can show: each consecutive pair of subwalks contains an arc served by a **distinct** vehicle tour of an **optimal solution (dotted).**



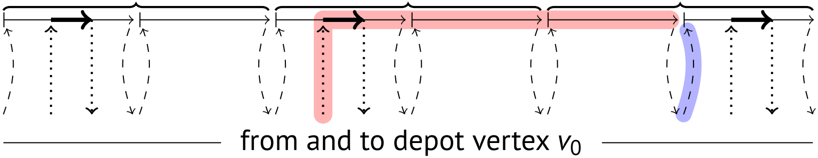
Charge length of the blue shortest paths to the red walks, consisting of arcs of the optimal solution and of arcs of the approximate MWRPP tour.

Each arc is **charged** to at most **five times** $\rightsquigarrow$ total cost of shortest paths is at most $5(\alpha(C+1)+3)\cdot \mathrm{OPT} + 5\cdot \mathrm{OPT}$.

Approximation factor analysis for MWCARP

Goal: Analyze total length of the **shortest paths (dashed)** added to the subwalks of the **MWRPP tour (solid)**.

One can show: each consecutive pair of subwalks contains an arc served by a **distinct** vehicle tour of an **optimal solution (dotted).**



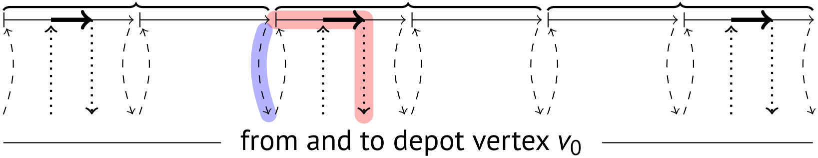
Charge length of the blue shortest paths to the red walks, consisting of arcs of the optimal solution and of arcs of the approximate MWRPP tour.

Each arc is **charged** to at most **five times** $\rightsquigarrow$ total cost of shortest paths is at most $5(\alpha(C+1)+3) \cdot \mathrm{OPT} + 5 \cdot \mathrm{OPT}$.

Approximation factor analysis for MWCARP

Goal: Analyze total length of the **shortest paths (dashed)** added to the subwalks of the **MWRPP tour (solid)**.

One can show: each consecutive pair of subwalks contains an arc served by a **distinct** vehicle tour of an **optimal solution (dotted).**



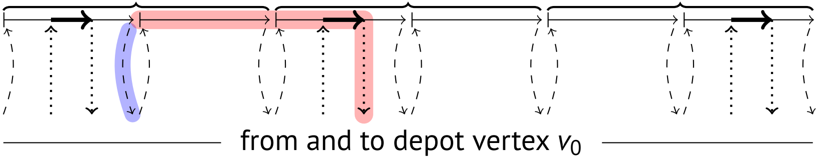
Charge length of the blue shortest paths to the red walks, consisting of arcs of the optimal solution and of arcs of the approximate MWRPP tour.

Each arc is **charged** to at most **five times** $\rightsquigarrow$ total cost of shortest paths is at most $5(\alpha(C+1)+3) \cdot \mathrm{OPT} + 5 \cdot \mathrm{OPT}$.

Approximation factor analysis for MWCARP

Goal: Analyze total length of the **shortest paths (dashed)** added to the subwalks of the **MWRPP tour (solid)**.

One can show: each consecutive pair of subwalks contains an arc served by a **distinct** vehicle tour of an **optimal solution (dotted).**



Charge length of the blue shortest paths to the red walks, consisting of arcs of the optimal solution and of arcs of the approximate MWRPP tour.

Each arc is **charged** to at most **five times** $\rightsquigarrow$ total cost of shortest paths is at most $5(\alpha(C+1)+3) \cdot \mathrm{OPT} + 5 \cdot \mathrm{OPT}$.

Conclusion

We have seen that

- MWCARP is not only as hard to approximate as $\triangle$-ATSP, but also how any improvement of $\triangle$-ATSP approximations transfers to MWCARP,

Conclusion

We have seen that

- MWCARP is not only as hard to approximate as $\triangle$-ATSP, but also how any improvement of $\triangle$-ATSP approximations transfers to MWCARP,
- MWCARP is constant-factor approximable in many practical cases.

Conclusion

We have seen that

- MWCARP is not only as hard to approximate as $\triangle$-ATSP, but also how any improvement of $\triangle$-ATSP approximations transfers to MWCARP,
- MWCARP is constant-factor approximable in many practical cases.

Future work:

- Implementation and algorithm engineering.

Conclusion

We have see that

- MWCARP is not only as hard to approximate as $\triangle$-ATSP, but also how any improvement of $\triangle$-ATSP approximations transfers to MWCARP,
- MWCARP is constant-factor approximable in many practical cases.

Future work:

- Implementation and algorithm engineering.
- Turn constraints, time windows, . . .