Polynomially solvable special cases of the Path-TSP

Eranda Çela, Graz University of Technology

joint work with

Vladimir Deineko, Warvick Business School

Gerhard J. Woeginger, RWTH Aachen

18th Cologne-Twente Workshop on Graphs and Combinatorial Optimization, September 14-16, 2020 (online)

Contents

Contents

The Path-TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\{1, 2, \ldots, n\} =: \overline{1..n}$,

The Path-TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\{1, 2, \ldots, n\} =: \overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$

The Path-TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\{1, 2, \ldots, n\} =: \overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$
an origin city s and a destination city t, $s < t$.

The Path-TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\{1, 2, \ldots, n\} =: \overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$
an origin city s and a destination city t, $s < t$.

Output: A path of minimum length which start at s, ends at t and visits each city exactly once.

The Path-TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\{1, 2, \ldots, n\} =: \overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$
an origin city s and a destination city t, $s < t$.

Output: A path of minimum length which start at s, ends at t and visits each city exactly once.

Notation: an (s, t)-TSP-path $\tau := \langle i_1, i_2, \ldots, i_n \rangle$, where $i_1 = s$, $i_n = t$ and $i_k \in \overline{1..n}$ for all k.

The Path-TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\{1, 2, \ldots, n\} =: \overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$
an origin city s and a destination city t, $s < t$.

Output: A path of minimum length which start at s, ends at t and visits each city exactly once.

Notation: an (s,t)-TSP-path $\tau := \langle i_1, i_2, \ldots, i_n \rangle$, where $i_1 = s$, $i_n = t$ and $i_k \in \overline{1..n}$ for all k.

$\tau(j)$ denotes the successor of a city j in τ.

The Path-TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\{1, 2, \ldots, n\} =: \overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$
an origin city s and a destination city t, $s < t$.

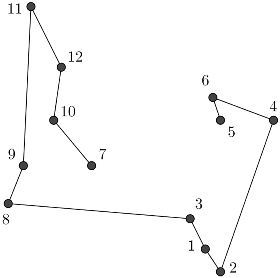
Output: A path of minimum length which start at s, ends at t and visits each city exactly once.

Notation: an (s, t)-TSP-path $\tau := \langle i_1, i_2, \ldots, i_n \rangle$, where $i_1 = s$, $i_n = t$ and $i_k \in \overline{1..n}$ for all k.

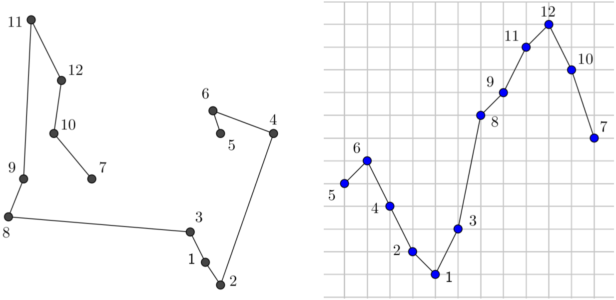
$\tau(j)$ denotes the successor of a city j in τ.

$\tau^{-1}(j)$ denotes the predecessor of a city j in τ.

A schematic representation of an (s, t)-TSP-path



A schematic representation of an (s, t)-TSP-path



Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\overline{1..n}$,

TSP

TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$

TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$

Output: A closed path of minimum length (=tour) which visits each city exactly once.

TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$

Output: A closed path of minimum length (=tour) which visits each city exactly once.

Path-TSP solvable in polynomial time $\Rightarrow$ TSP solvable in polynomial time

TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$

Output: A closed path of minimum length (=tour) which visits each city exactly once.

Path-TSP solvable in polynomial time $\Rightarrow$ TSP solvable in polynomial time

Thus **Path-TSP is NP-hard**.

TSP

Input:	Size $n \in \mathbb{N}$ of the problem, a set of n cities $\overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$
Output:	A closed path of minimum length (=tour) which visits each city exactly once.

Path-TSP solvable in polynomial time $\Rightarrow$ TSP solvable in polynomial time

Thus **Path-TSP is NP-hard**.

TSP and Path-TSP are APX-hard (Papadimitriou at al. '93, Karpinski et al. 2015)

TSP

Input: Size $n \in \mathbb{N}$ of the problem, a set of n cities $\overline{1..n}$, a symmetric nonnegative matrix $C = (c_{ij})$ of distances c_{ij} between cities i and j, $\forall i, j \in \overline{1..n}$

Output: A closed path of minimum length (=tour) which visits each city exactly once.

Path-TSP solvable in polynomial time $\Rightarrow$ TSP solvable in polynomial time

Thus **Path-TSP is NP-hard**.

TSP and Path-TSP are APX-hard (Papadimitriou at al. '93, Karpinski et al. 2015)

Recently: the approximability of Path-TSP is the same as for TSP, up to an arbitrarily small error. (Traub et al. 2019)

Introduction: Polynomially solvable special cases (p.s.s.c.)

A p.s.s.c. is characterised in terms of some property of the input; if the input fulfills that property, the considered problem is polynomially solvable.

A p.s.s.c. is characterised in terms of some property of the input; if the input fulfills that property, the considered problem is polynomially solvable.

Many p.s.s.c. known for the TSP:

A p.s.s.c. is characterised in terms of some property of the input; if the input fulfills that property, the considered problem is polynomially solvable.

Many p.s.s.c. known for the TSP:
The particular properties of the input are often represented in terms of inequalities fulfilled by the entries of the distance matrix C
(Monge matrices, Kalmanson matrices, Demidenko matrices, Van der Veen matrices,...)
(Gilmore et al. 1985, Burkard et al. 1998, Kabadi 2007, Deineko et al. 2014)

A p.s.s.c. is characterised in terms of some property of the input; if the input fulfills that property, the considered problem is polynomially solvable.

Many p.s.s.c. known for the TSP:
The particular properties of the input are often represented in terms of inequalities fulfilled by the entries of the distance matrix C
(Monge matrices, Kalmanson matrices, Demidenko matrices, Van der Veen matrices,...)
(Gilmore et al. 1985, Burkard et al. 1998, Kabadi 2007, Deineko et al. 2014)

Just one p.s.s.c. of the Path-TSP known:
the cities are points in the Eucledean plane and all cities lie on the boundary of their convex hull. (Garcia et al. 1996, 1998)

A p.s.s.c. is characterised in terms of some property of the input; if the input fulfills that property, the considered problem is polynomially solvable.

Many p.s.s.c. known for the TSP:
The particular properties of the input are often represented in terms of inequalities fulfilled by the entries of the distance matrix C
(Monge matrices, Kalmanson matrices, Demidenko matrices, Van der Veen matrices,...)
(Gilmore et al. 1985, Burkard et al. 1998, Kabadi 2007, Deineko et al. 2014)

Just one p.s.s.c. of the Path-TSP known:
the cities are points in the Eucledean plane and all cities lie on the boundary of their convex hull. (Garcia et al. 1996, 1998)

More generally, **if C is a Kalmanson matrix, then Path-TSP is solvable in $O(n)$ time.**

A p.s.s.c. is characterised in terms of some property of the input; if the input fulfills that property, the considered problem is polynomially solvable.

Many p.s.s.c. known for the TSP:
The particular properties of the input are often represented in terms of inequalities fulfilled by the entries of the distance matrix C (Monge matrices, Kalmanson matrices, Demidenko matrices, Van der Veen matrices,...)
(Gilmore et al. 1985, Burkard et al. 1998, Kabadi 2007, Deineko et al. 2014)

Just one p.s.s.c. of the Path-TSP known:
the cities are points in the Eucledean plane and all cities lie on the boundary of their convex hull. (Garcia et al. 1996, 1998)

More generally, **if C is a Kalmanson matrix, then Path-TSP is solvable in $O(n)$ time.**

In general p.s.s.c. of TSP do not translate to p.s.s.c. of Path-TSP!

A p.s.s.c. is characterised in terms of some property of the input; if the input fulfills that property, the considered problem is polynomially solvable.

Many p.s.s.c. known for the TSP:
The particular properties of the input are often represented in terms of inequalities fulfilled by the entries of the distance matrix C (Monge matrices, Kalmanson matrices, Demidenko matrices, Van der Veen matrices,...)
(Gilmore et al. 1985, Burkard et al. 1998, Kabadi 2007, Deineko et al. 2014)

Just one p.s.s.c. of the Path-TSP known:
the cities are points in the Eucledean plane and all cities lie on the boundary of their convex hull. (Garcia et al. 1996, 1998)

More generally, **if C is a Kalmanson matrix, then Path-TSP is solvable in $O(n)$ time.**

In general p.s.s.c. of TSP do not translate to p.s.s.c. of Path-TSP!

Contents

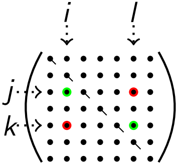
Demidenko matrices and pyramidal paths

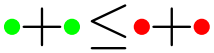
Demidenko matrices and pyramidal paths

A symmetric $n \times n$ matrix $C = (c_{ij})$ is called a Demidenko matrix if $c_{ji} + c_{kl} \leq c_{jl} + c_{ki}$, for all $1 \leq i < j < k < l \leq n$.

Demidenko matrices and pyramidal paths

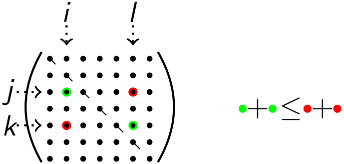
A symmetric $n \times n$ matrix $C = (c_{ij})$ is called a Demidenko matrix if $c_{ji} + c_{kl} \leq c_{jl} + c_{ki}$, for all $1 \leq i < j < k < l \leq n$.





Demidenko matrices and pyramidal paths

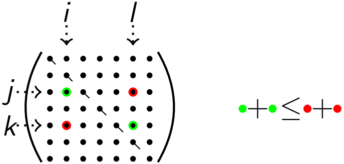
A symmetric $n \times n$ matrix $C = (c_{ij})$ is called a Demidenko matrix if $c_{ji} + c_{kl} \leq c_{jl} + c_{ki}$, for all $1 \leq i < j < k < l \leq n$.



Demidenko matrices are a strict superclass of Kalmanson matrices

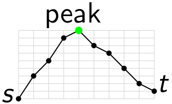
Demidenko matrices and pyramidal paths

A symmetric $n \times n$ matrix $C = (c_{ij})$ is called a Demidenko matrix if $c_{ji} + c_{kl} \leq c_{jl} + c_{ki}$, for all $1 \leq i < j < k < l \leq n$.



Demidenko matrices are a strict superclass of Kalmanson matrices

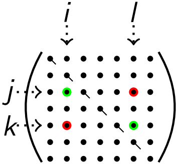
λ-pyramidal path

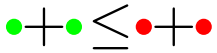


The Path-TSP with a Demidenko distance matrix

Demidenko matrices and pyramidal paths

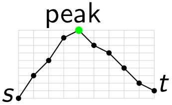
A symmetric $n \times n$ matrix $C = (c_{ij})$ is called a Demidenko matrix if $c_{ji} + c_{kl} \leq c_{jl} + c_{ki}$, for all $1 \leq i < j < k < l \leq n$.



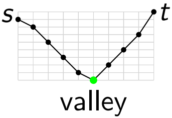


Demidenko matrices are a strict superclass of Kalmanson matrices

λ-pyramidal path



ν-pyramidal path



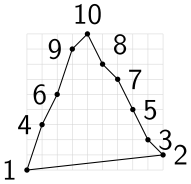
Demidenko matrices and pyramidal tours

Demidenko matrices and pyramidal tours

A closed λ-pyramidal or ν-pyramidal path is called a pyramidal tour.

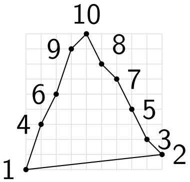
Demidenko matrices and pyramidal tours

A closed λ-pyramidal or ν-pyramidal path is called a pyramidal tour.



Demidenko matrices and pyramidal tours

A closed λ-pyramidal or ν-pyramidal path is called a pyramidal tour.



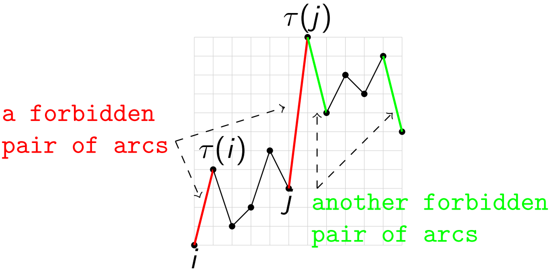
Theorem (Demidenko, 1979)
For a set of cities with a Demidenko distance matrix there exists an optimal tour which is pyramidal. In this case the TSP on n cities can be solved in $O(n^2)$ time by dynamical programming.

Forbidden pairs of arcs

In an (s,t)-path τ a pair of arcs $(i,\tau(i))$ and $(j,\tau(j))$ is called a forbidden pair of arcs if either $i<j<\tau(i)<\tau(j)$ or $i>j>\tau(i)>\tau(j)$ holds.

Forbidden pairs of arcs

In an (s,t)-path τ a pair of arcs $(i,\tau(i))$ and $(j,\tau(j))$ is called a forbidden pair of arcs if either $i < j < \tau(i) < \tau(j)$ or $i > j > \tau(i) > \tau(j)$ holds.



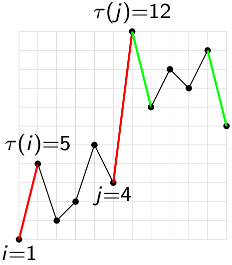
The forbidden pairs lemma: In the case of a Demidenko distance matrix there always exist an optimal (s,t)-TSP-path without forbidden pairs of arcs, for any s and t.

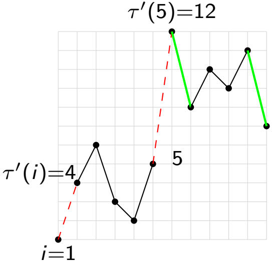
Proof of the forbidden pairs lemma

Exchange argument: replace the forbidden pair of arcs $(i, \tau(i))$, $(j, \tau(j))$ in an optimal path by the non-forbidden pair (i, j), $(\tau(i), \tau(j))$ and reverse the $(\tau(i), j)$-subpath. The resulting path is again optimal.

Proof of the forbidden pairs lemma

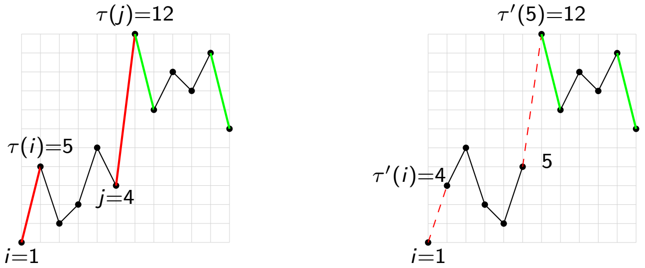
Exchange argument: replace the forbidden pair of arcs $(i, \tau(i))$, $(j, \tau(j))$ in an optimal path by the non-forbidden pair (i, j), $(\tau(i), \tau(j))$ and reverse the $(\tau(i), j)$-subpath. The resulting path is again optimal.





Proof of the forbidden pairs lemma

Exchange argument: replace the forbidden pair of arcs $(i, \tau(i))$, $(j, \tau(j))$ in an optimal path by the non-forbidden pair (i, j), $(\tau(i), \tau(j))$ and reverse the $(\tau(i), j)$-subpath. The resulting path is again optimal.



After repeating the exchange argument a finite number of times we get an optimal (s, t)-TSP-path without forbidden pairs of arcs.

The special structure of optimal paths

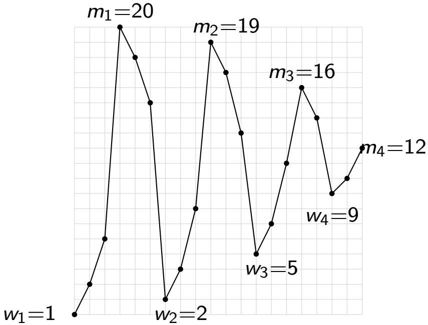
The special structure of optimal paths

Lemma 1: $\langle 1, 2, \ldots, n \rangle$ is a shortest $(1, n)$-TSP-path.

The special structure of optimal paths

Lemma 1: $\langle 1, 2, \ldots, n \rangle$ is a shortest $(1, n)$-TSP-path.

Lemma 2: For any t there exist an optimal $(1, t)$-TSP-path without forbidden pairs of arcs in which the peaks decrease and the valleys increase along the path.



The special structure of optimal paths

The special structure of optimal paths

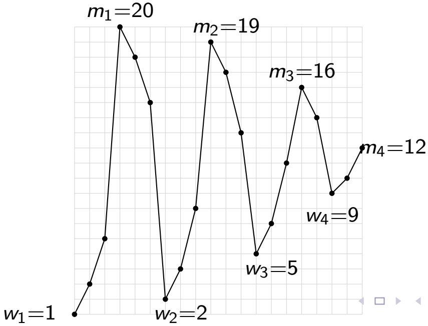
Lemma 3: In an optimal $(1,t)$-TSP-path without forbidden pairs of arcs and with decreasing peaks and increasing valleys, the following holds for any alternating sequence of valleys and peaks w_1, m_1, w_2, m_2, w_3:

The special structure of optimal paths

Lemma 3: In an optimal $(1,t)$-TSP-path without forbidden pairs of arcs and with decreasing peaks and increasing valleys, the following holds for any alternating sequence of valleys and peaks w_1,m_1,w_2,m_2,w_3:

(I) The (w_1,m_1)-subpath contains no indices i such that $w_2 < i < m_2$.

(II) The (m_1,w_2)-subpath contains no indices j such that $w_3 < j < m_2$.

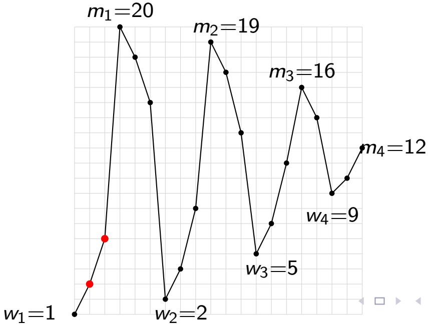


The special structure of optimal paths

Lemma 3: In an optimal $(1, t)$-TSP-path without forbidden pairs of arcs and with decreasing peaks and increasing valleys, the following holds for any alternating sequence of valleys and peaks w_1,m_1,w_2,m_2,w_3:

(I) The (w_1, m_1)-subpath contains no indices i such that $w_2 < i < m_2$.

(II) The (m_1, w_2)-subpath contains no indices j such that $w_3 < j < m_2$.

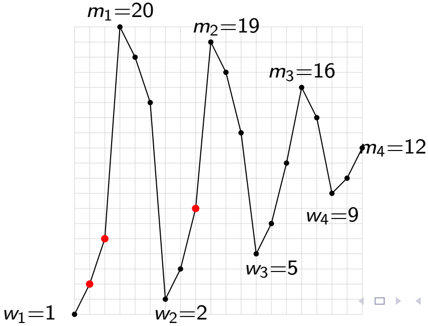


The special structure of optimal paths

Lemma 3: In an optimal $(1, t)$-TSP-path without forbidden pairs of arcs and with decreasing peaks and increasing valleys, the following holds for any alternating sequence of valleys and peaks w_1,m_1,w_2,m_2,w_3:

(I) The (w_1, m_1)-subpath contains no indices i such that $w_2 < i < m_2$.

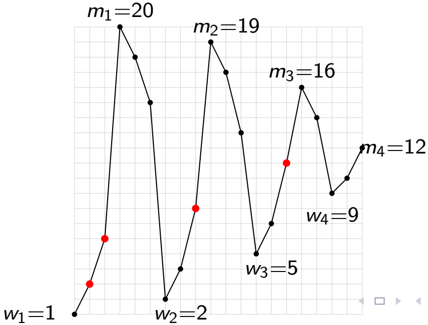
(II) The (m_1, w_2)-subpath contains no indices j such that $w_3 < j < m_2$.



The special structure of optimal paths

Lemma 3: In an optimal $(1,t)$-TSP-path without forbidden pairs of arcs and with decreasing peaks and increasing valleys, the following holds for any alternating sequence of valleys and peaks w_1,m_1,w_2,m_2,w_3:

(I) The (w_1,m_1)-subpath contains no indices i such that $w_2 < i < m_2$.
(II) The (m_1,w_2)-subpath contains no indices j such that $w_3 < j < m_2$.

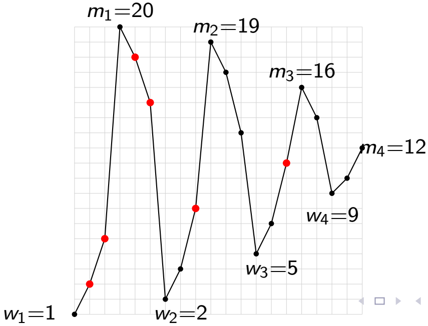


The special structure of optimal paths

Lemma 3: In an optimal $(1, t)$-TSP-path without forbidden pairs of arcs and with decreasing peaks and increasing valleys, the following holds for any alternating sequence of valleys and peaks w_1, m_1, w_2, m_2, w_3:

(I) The (w_1, m_1)-subpath contains no indices i such that $w_2 < i < m_2$.

(II) The (m_1, w_2)-subpath contains no indices j such that $w_3 < j < m_2$.

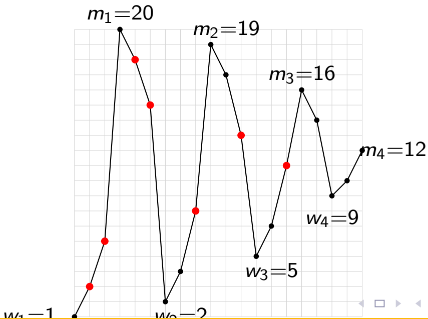


The special structure of optimal paths

Lemma 3: In an optimal $(1,t)$-TSP-path without forbidden pairs of arcs and with decreasing peaks and increasing valleys, the following holds for any alternating sequence of valleys and peaks w_1, m_1, w_2, m_2, w_3:

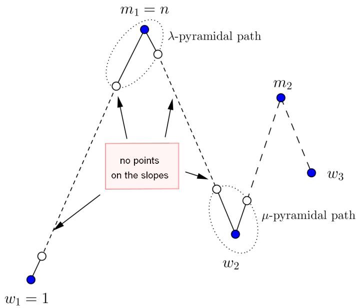
(I) The (w_1, m_1)-subpath contains no indices i such that $w_2 < i < m_2$.

(II) The (m_1, w_2)-subpath contains no indices j such that $w_3 < j < m_2$.

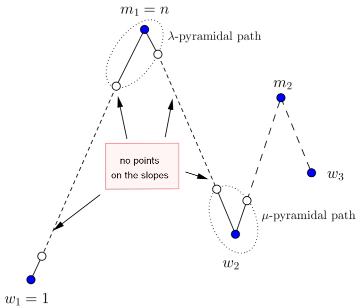


The special structure of optimal paths

The special structure of optimal paths



The special structure of optimal paths



Theorem: For every destination city t the $(1, t)$-Path-TSP on n cities with a Demidenko distance matrix can be solved in $O(n^5)$ time by dynamic programming.

Structural properties

Structural properties

For $1 < s < t < n$ there is an optimal (s, t)-TSP-path τ without forbidden pairs of arcs, where

Structural properties

For $1 < s < t < n$ there is an optimal (s, t)-TSP-path τ without forbidden pairs of arcs, where

(A) 1 precedes n in τ,

Structural properties

For $1 < s < t < n$ there is an optimal (s,t)-TSP-path τ without forbidden pairs of arcs, where

(A) 1 precedes n in τ,

(B) $i < j$ holds for each city i in the $(s,1)$-subpath and for each city j in the (n,t)-subpath,

Structural properties

For $1 < s < t < n$ there is an optimal (s,t)-TSP-path τ without forbidden pairs of arcs, where

(A) 1 precedes n in τ,

(B) $i < j$ holds for each city i in the $(s,1)$-subpath and for each city j in the (n,t)-subpath,

(C) There exists a city p such that $\tau = \langle \tau_1^p, \tau_2^p \rangle$, where τ_1^p visits all cities in $\overline{1, p-1}$ and τ_2^p visits all cities in $\overline{p, n}$.

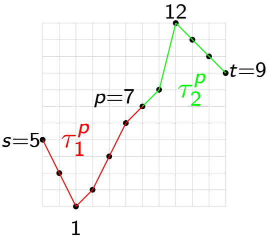
Structural properties

For $1 < s < t < n$ there is an optimal (s,t)-TSP-path τ without forbidden pairs of arcs, where

(A) 1 precedes n in τ,

(B) $i < j$ holds for each city i in the $(s,1)$-subpath and for each city j in the (n,t)-subpath,

(C) There exists a city p such that $\tau = \langle \tau_1^p, \tau_2^p \rangle$, where τ_1^p visits all cities in $\overline{1, p-1}$ and τ_2^p visits all cities in $\overline{p, n}$.



Theorem: The (s,t)-Path-TSP on n cities with a Demidenko distance matrix can be solved in $O((t-s)n^5)$ time for any origin city s and any destination city t, $1 < s < t < n$.

Contents

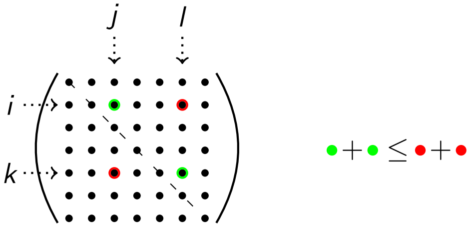
Van der Veen matrices and pyramidal tours

Van der Veen matrices and pyramidal tours

Definition: A symmetric matrix $C = (c_{ij})$ is called a Van der Veen matrix if $c_{ij} + c_{kl} \leq c_{il} + c_{kj}$, for all $1 \leq i < j < k < l \leq n$.

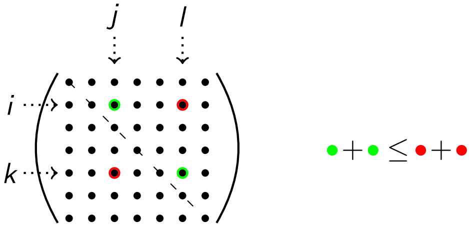
Van der Veen matrices and pyramidal tours

Definition: A symmetric matrix $C = (c_{ij})$ is called a Van der Veen matrix if $c_{ij} + c_{kl} \leq c_{il} + c_{kj}$, for all $1 \leq i < j < k < l \leq n$.



Van der Veen matrices and pyramidal tours

Definition: A symmetric matrix $C = (c_{ij})$ is called a Van der Veen matrix if $c_{ij} + c_{kl} \leq c_{il} + c_{kj}$, for all $1 \leq i < j < k < l \leq n$.



Theorem (Van der Veen, 1994)
For a set of cities with a Van der Veen distance matrix, there exists an optimal tour which is pyramidal. In this case the TSP on n cities can be solved in $O(n^2)$ time by dynamical programming.

Forbidden pairs of arcs

In an (s,t)-path τ a pair of arcs $(i,\tau(i))$ and $(j,\tau(j))$ is called a forbidden pair of arcs of type A or B if the condition (A) or (B) holds

(A) $i < j < \tau(j) < \tau(i)$ or $i > j > \tau(j) > \tau(i)$

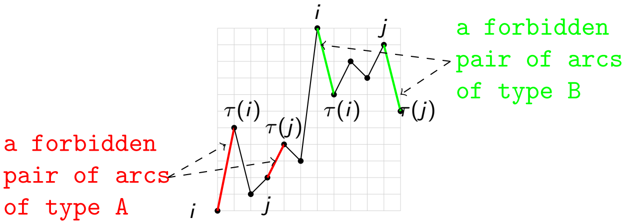
(B) $i < j < j+1 < \tau(i) < \tau(j)$ or $\tau(j) < \tau(i) < \tau(i)+1 < j < i$

Forbidden pairs of arcs

In an (s,t)-path τ a pair of arcs $(i,\tau(i))$ and $(j,\tau(j))$ is called a forbidden pair of arcs of type A or B if the condition (A) or (B) holds

(A) $i < j < \tau(j) < \tau(i)$ or $i > j > \tau(j) > \tau(i)$

(B) $i < j < j+1 < \tau(i) < \tau(j)$ or $\tau(j) < \tau(i) < \tau(i)+1 < j < i$

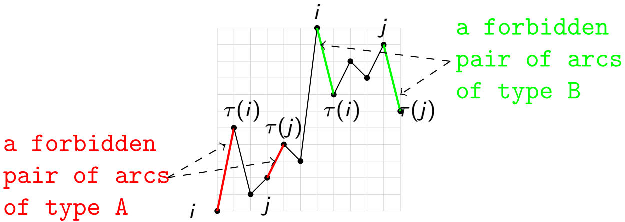


Forbidden pairs of arcs

In an (s,t)-path τ a pair of arcs $(i,\tau(i))$ and $(j,\tau(j))$ is called a forbidden pair of arcs of type A or B if the condition (A) or (B) holds

(A) $i < j < \tau(j) < \tau(i)$ or $i > j > \tau(j) > \tau(i)$

(B) $i < j < j+1 < \tau(i) < \tau(j)$ or $\tau(j) < \tau(i) < \tau(i)+1 < j < i$



The forbidden pairs lemma: In the case of a Van der Veen distance matrix (c_{ij}), there is an optimal (s,t)-TSP-path without forbidden pairs of arcs for any s and t.

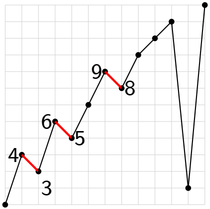
Particular subpaths

Particular subpaths

An arc $(k+1, k)$ in a path $\tau = \langle \ldots, i, k+1, k, j, \ldots \rangle$ with $i = \tau^{-1}(k+1) < \tau(k+1) = k < k+1 < \tau(k) = j$ is called a short zigzag with peak $k+1$ (and valley k).

Particular subpaths

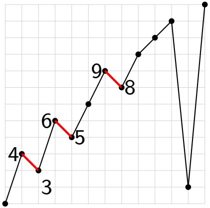
An arc $(k+1, k)$ in a path $\tau = \langle \ldots, i, k+1, k, j, \ldots \rangle$ with $i = \tau^{-1}(k+1) < \tau(k+1) = k < k+1 < \tau(k) = j$ is called a short zigzag with peak $k+1$ (and valley k).



the red arcs are short zigzags

Particular subpaths

An arc $(k+1,k)$ in a path $\tau = \langle \dots, i, k+1, k, j, \dots \rangle$ with $i = \tau^{-1}(k+1) < \tau(k+1) = k < k+1 < \tau(k) = j$ is called a short zigzag with peak $k+1$ (and valley k).



the red arcs are short zigzags

Short zigzag lemma: Let τ be a $(1,n)$-TSP-path without forbidden pairs of arcs such that $\tau(k+1) = k$ for some $k \in \overline{1, n-1}$. Then $(k+1,k)$ is a short zigzag with peak $k+1$ and $\tau(k) \in \{k+2, k+3\}$, $\tau^{-1}(k+1) \in \{k-2, k-1\}$ hold.

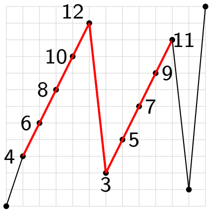
Particular subpaths

Particular subpaths

A path $\tau_{p,v} = \langle v+1, v+3, \dots, p-2, p, v, v+2, \dots, p-1\rangle$ with $v < p-1$, and v, p of different parity, $v, p \in \overline{1, n}$, is called a long zigzag with peak p and valley v and is denoted by $\tau_{p,v}$.

Particular subpaths

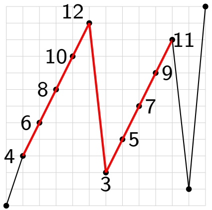
A path $\tau_{p,v} = \langle v+1, v+3, \ldots, p-2, p, v, v+2, \ldots, p-1 \rangle$ with $v < p-1$, and v, p of different parity, $v, p \in \overline{1, n}$, is called a long zigzag with peak p and valley v and is denoted by $\tau_{p,v}$.



the red subpath is a long zigzag $\tau_{12,3}$ with peak 12 and valley 3

Particular subpaths

A path $\tau_{p,v} = \langle v+1, v+3, \ldots, p-2, p, v, v+2, \ldots, p-1\rangle$ with $v < p-1$, and v, p of different parity, $v, p \in \overline{1, n}$, is called a long zigzag with peak p and valley v and is denoted by $\tau_{p,v}$.



the red subpath is a long zigzag $\tau_{12,3}$ with peak 12 and valley 3

Long zigzag lemma: Let τ be a $(1, n)$-TSP-path without forbidden pairs of arcs. Let p be a peak and let $v < p-1$ be the first valley reached after p in τ. Then the long zigzag $\tau_{p,v}$ is a subpath of τ. Moreover $\tau_{p,v}$ is preceeded by $x \in \{v-1, v-2\}$ and is succeeded by $y \in \{p+1, p+2\}$.

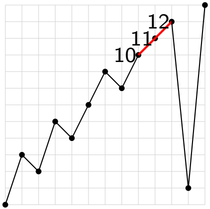
Particular subpaths

Particular subpaths

For two cities i, j with $i, j \in \overline{0, n+1}$, $i < j - 1$ the monotone increasing subpath $\langle i+1, \ldots, j-1 \rangle$ is called a chain between i and j and is denoted by $ch(i, j)$.

Particular subpaths

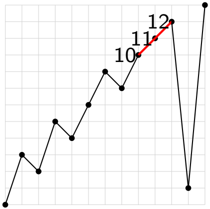
For two cities i, j with $i, j \in \overline{0, n+1}$, $i < j - 1$ the monotone increasing subpath $\langle i+1, \ldots, j-1 \rangle$ is called a chain between i and j and is denoted by $ch(i, j)$.



the red subpath is
the chain $ch(9, 13)$

Particular subpaths

For two cities i, j with $i, j \in \overline{0, n+1}$, $i < j - 1$ the monotone increasing subpath $\langle i+1, \ldots, j-1 \rangle$ is called a chain between i and j and is denoted by $ch(i, j)$.



the red subpath is the chain $ch(9, 13)$

Zigzag paths

Zigzag paths

A $(1, n)$-TSP-path τ is called a zigzag path if

(I) τ is monotone increasing

Zigzag paths

A $(1, n)$-TSP-path τ is called a zigzag path if

(I) τ is monotone increasing

(II) τ is a sequence of long and/or short zigzags z_1, z_2, ..., z_k with peaks p_i and valley v_i, for $i \in \overline{1, k}$, such that $1 < v_1 < p_1 < v_2 < p_2 < \ldots < v_k < p_k < n$.

Zigzag paths

A $(1, n)$-TSP-path τ is called a zigzag path if

(I) τ is monotone increasing

(II) τ is a sequence of long and/or short zigzags $z_1, z_2, \ldots, z_k$ with peaks p_i and valley v_i, for $i \in \overline{1, k}$, such that $1 < v_1 < p_1 < v_2 < p_2 < \ldots < v_k < p_k < n$.

τ starts with the chain $ch(0, v_1)$ and ends with the chain $ch(p_k, n+1)$.

Zigzag paths

A $(1, n)$-TSP-path τ is called a zigzag path if

(I) τ is monotone increasing

(II) τ is a sequence of long and/or short zigzags $z_1, z_2, \ldots, z_k$ with peaks p_i and valley v_i, for $i \in \overline{1, k}$, such that
$1 < v_1 < p_1 < v_2 < p_2 < \ldots < v_k < p_k < n$.
τ starts with the chain $ch(0, v_1)$ and ends with the chain $ch(p_k, n+1)$.
Two consecutive zigzags z_i and z_{i+1} for which $p_i = v_{i+1} - 1$ holds, are joined by the edge $(p_i - 1, p_i + 2)$.

Zigzag paths

A $(1, n)$-TSP-path τ is called a zigzag path if

(I) τ is monotone increasing

(II) τ is a sequence of long and/or short zigzags $z_1, z_2, \ldots, z_k$ with peaks p_i and valley v_i, for $i \in \overline{1, k}$, such that $1 < v_1 < p_1 < v_2 < p_2 < \ldots < v_k < p_k < n$.

τ starts with the chain $ch(0, v_1)$ and ends with the chain $ch(p_k, n+1)$.

Two consecutive zigzags z_i and z_{i+1} for which $p_i = v_{i+1} - 1$ holds, are joined by the edge $(p_i - 1, p_i + 2)$.

Otherwise, if $p_i < v_{i+1} - 1$, the zigzags z_i and z_{i+1} are connected by the chain $ch(p_i, v_{i+1})$.

Zigzag paths

A $(1, n)$-TSP-path τ is called a zigzag path if

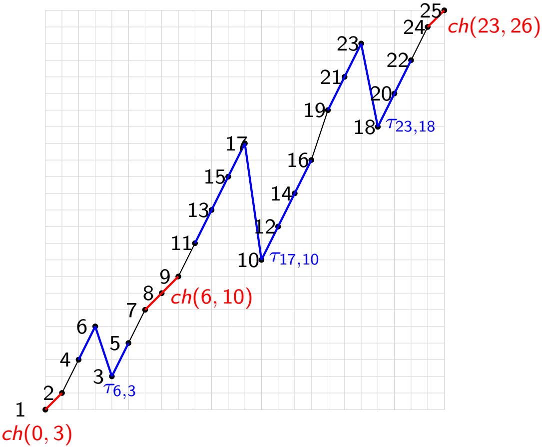
(I) τ is monotone increasing

(II) τ is a sequence of long and/or short zigzags z_1, z_2, ..., z_k with peaks p_i and valley v_i, for $i \in \overline{1, k}$, such that
$1 < v_1 < p_1 < v_2 < p_2 < \ldots < v_k < p_k < n$.
τ starts with the chain $ch(0, v_1)$ and ends with the chain $ch(p_k, n + 1)$.
Two consecutive zigzags z_i and z_{i+1} for which $p_i = v_{i+1} - 1$ holds, are joined by the edge $(p_i - 1, p_i + 2)$.
Otherwise, if $p_i < v_{i+1} - 1$, the zigzags z_i and z_{i+1} are connected by the chain $ch(p_i, v_{i+1})$.

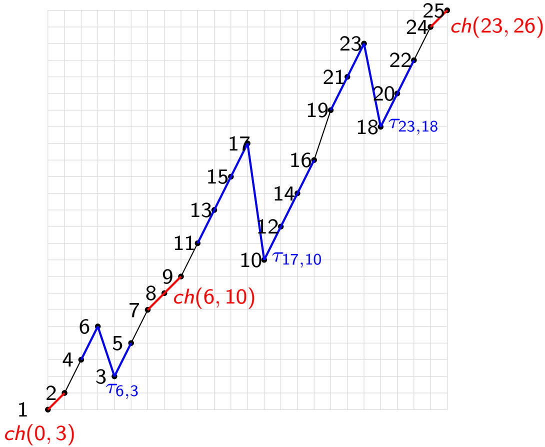
Zigzag path lemma: Any $(1, n)$-TSP-path without forbidden pairs of arcs is a zigzag path.

Example of a zigzag path

Example of a zigzag path



Example of a zigzag path



Modelling zigzag paths on a directed graph with $O(n)$ vertices

Modelling zigzag paths on a directed graph with $O(n)$ vertices

The vertex set: $V_n := \{1, 2, \ldots, n\} \cup \{\bar{3}, \bar{4}, \ldots, \overline{n-1}\} \cup \{\underline{2}, \underline{3}, \ldots, \underline{n-2}\}$.

Modelling zigzag paths on a directed graph with $O(n)$ vertices

The vertex set: $V_n := \{1, 2, \ldots, n\} \cup \{\bar{3}, \bar{4}, \ldots, \overline{n-1}\} \cup \{\underline{2}, \underline{3}, \ldots, \underline{n-2}\}$.
The set of arcs: $E_n := E_n^{(1)} \cup E_n^{(2)} \cup E_n^{(3)} \cup E_n^{(4)}$, where

- $E_n^{(1)}$ represents arcs belonging to chains;

Modelling zigzag paths on a directed graph with $O(n)$ vertices

The vertex set: $V_n := \{1, 2, \ldots, n\} \cup \{\overline{3}, \overline{4}, \ldots, \overline{n-1}\} \cup \{\underline{2}, \underline{3}, \ldots, \underline{n-2}\}$.
The set of arcs: $E_n := E_n^{(1)} \cup E_n^{(2)} \cup E_n^{(3)} \cup E_n^{(4)}$, where

- $E_n^{(1)}$ represents arcs belonging to chains;
- $E_n^{(2)}$ represents the short zigzags, $(\overline{i+1}, \underline{i})$ represents $(i+1, i)$;

Modelling zigzag paths on a directed graph with $O(n)$ vertices

The vertex set: $V_n := \{1, 2, \ldots, n\} \cup \{\bar{3}, \bar{4}, \ldots, \overline{n-1}\} \cup \{\underline{2}, \underline{3}, \ldots, \underline{n-2}\}$.
The set of arcs: $E_n := E_n^{(1)} \cup E_n^{(2)} \cup E_n^{(3)} \cup E_n^{(4)}$, where

- $E_n^{(1)}$ represents arcs belonging to chains;
- $E_n^{(2)}$ represents the short zigzags, $(\overline{i+1}, \underline{i})$ represents $(i+1, i)$;
- $E_n^{(3)}$ represents the long zigzags; $(\bar{i}, \underline{j})$ represents $\tau_{j+1,i-1}$ with peak $j+1$ and valley $i-1$;

Modelling zigzag paths on a directed graph with $O(n)$ vertices

The vertex set: $V_n := \{1, 2, \ldots, n\} \cup \{\overline{3}, \overline{4}, \ldots, \overline{n-1}\} \cup \{\underline{2}, \underline{3}, \ldots, \underline{n-2}\}$.
The set of arcs: $E_n := E_n^{(1)} \cup E_n^{(2)} \cup E_n^{(3)} \cup E_n^{(4)}$, where

- $E_n^{(1)}$ represents arcs belonging to chains;
- $E_n^{(2)}$ represents the short zigzags, $(\overline{i+1}, \underline{i})$ represents $(i+1, i)$;
- $E_n^{(3)}$ represents the long zigzags; $(\overline{i}, \underline{j})$ represents $\tau_{j+1,i-1}$ with peak $j+1$ and valley $i-1$;
- $E_n^{(4)}$ represents the arcs connecting a zigzag to the chain or the zigzag preceding or succeeding it;

Modelling zigzag paths on a directed graph with $O(n)$ vertices

The vertex set: $V_n := \{1, 2, \dots, n\} \cup \{\overline{3}, \overline{4}, \dots, \overline{n-1}\} \cup \{\underline{2}, \underline{3}, \dots, \underline{n-2}\}$.
The set of arcs: $E_n := E_n^{(1)} \cup E_n^{(2)} \cup E_n^{(3)} \cup E_n^{(4)}$, where

- $E_n^{(1)}$ represents arcs belonging to chains;
- $E_n^{(2)}$ represents the short zigzags, $\left(\overline{i+1}, \underline{i}\right)$ represents $(i+1, i)$;
- $E_n^{(3)}$ represents the long zigzags; $\left(\overline{i}, \underline{j}\right)$ represents $\tau_{j+1,i-1}$ with peak $j+1$ and valley $i-1$;
- $E_n^{(4)}$ represents the arcs connecting a zigzag to the chain or the zigzag preceding or succeeding it;
  - from a zigzag with peak $k+1$ to a zigzag with valley $k+2$: $\left(\underline{k}, \overline{k+3}\right)$
  - from a zigzag with peak $k+1$ to a chain: $\left(\underline{k}, k+2\right)$
  - from a chain to a zigzag with valley $k+1$: $\left(k, \overline{k+2}\right)$

Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

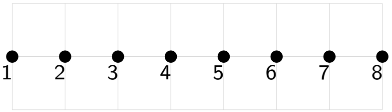
An example for $n = 8$

Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

An example for $n = 8$

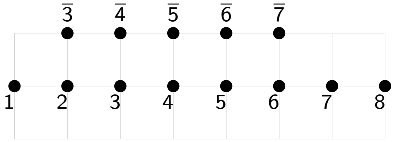


Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

An example for $n = 8$

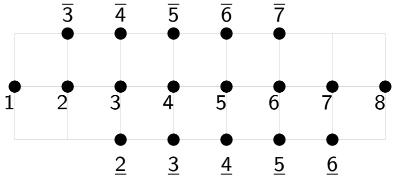


Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1,n)$ paths in the auxiliary graph.

Theorem: The shortest $(1,n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1,n}$ a family of shortest $(1,k)$-TSP-path over the cities $\overline{1,k}$ can be found in $O(n^2)$ time.

An example for $n = 8$

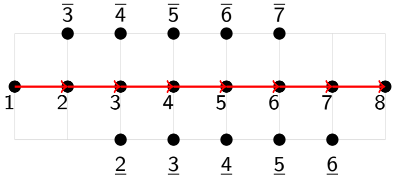


Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

An example for $n = 8$



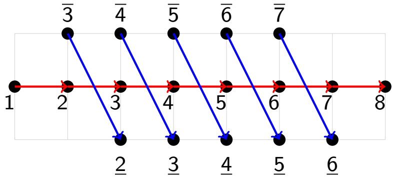
arcs contained in chains

Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

An example for $n = 8$



arcs contained in chains

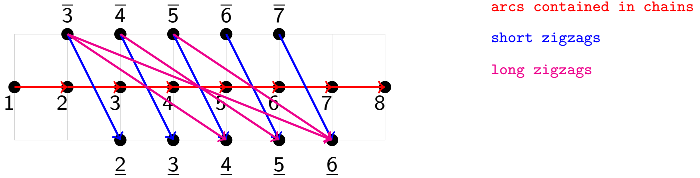
short zigzags

Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

An example for $n = 8$

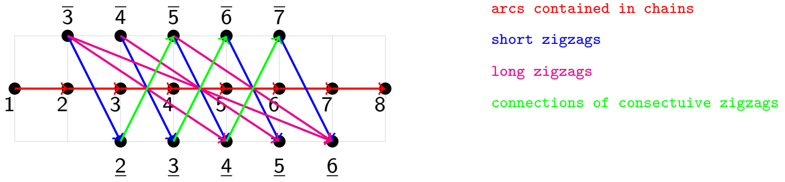


Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

An example for $n = 8$

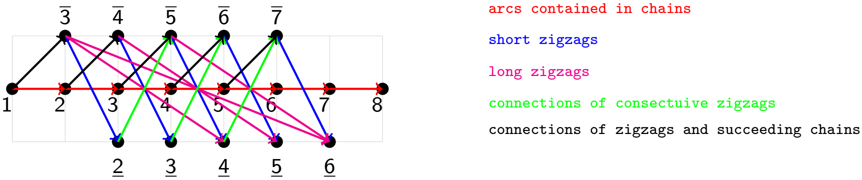


Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

An example for $n = 8$

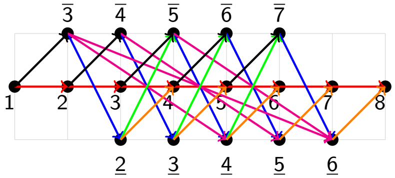


Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1, n)$ paths in the auxiliary graph.

Theorem: The shortest $(1, n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1, n}$ a family of shortest $(1, k)$-TSP-path over the cities $\overline{1, k}$ can be found in $O(n^2)$ time.

An example for $n = 8$



arcs contained in chains

short zigzags

long zigzags

connections of consecutuive zigzags

connections of zigzags and succeeding chains

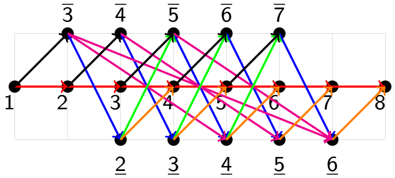
connections of chains and succeeding zigzags

Modelling zigzag paths on a directed graph with $O(n)$ vertices

Observation: The zigzag paths coincide with the $(1,n)$ paths in the auxiliary graph.

Theorem: The shortest $(1,n)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time. Moreover, for each $k \in \overline{1,n}$ a family of shortest $(1,k)$-TSP-path over the cities $\overline{1,k}$ can be found in $O(n^2)$ time.

An example for $n = 8$



arcs contained in chains

short zigzags

long zigzags

connections of consecutuive zigzags

connections of zigzags and succeeding chains

connections of chains and succeeding zigzags

The general case

The general case

Theorem: For any $t \in \{2, \ldots, n\}$ the shortest $(1, t)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time.

The general case

Theorem: For any $t \in \{2, \ldots, n\}$ the shortest $(1, t)$-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^2)$ time.

Theorem: For any $s, t \in \{1, 2, \ldots, n\}$, $1 < s < t < n$, a shortest (s, t)-TSP-path on n cities with a Van der Veen distance matrix can be found in $O(n^3)$ time.

Conclusions

- We discussed two new polynomially solvale special cases of the Path-TSP:
  (I) n cities with a Demidenko distance matrix: solvable in $(t-s)n^5$ for general s and t, solvable in $O(n^5)$ for $s=1$, solvable in $O(n)$ for $s=1$, $t=n$.
  (II) n cities with a Van der Veen distance matrix: solvable in $O(n^3)$ for general s and t, solvable in $O(n^2)$ if $s=1$.

Conclusions

- We discussed two new polynomially solvale special cases of the Path-TSP:
  (I) n cities with a Demidenko distance matrix: solvable in $(t-s)n^5$ for general s and t, solvable in $O(n^5)$ for $s=1$, solvable in $O(n)$ for $s=1$, $t=n$.
  (II) n cities with a Van der Veen distance matrix: solvable in $O(n^3)$ for general s and t, solvable in $O(n^2)$ if $s=1$.
- The optimal (s,t)-TSP-paths can be found among paths with a particular combinatorial structure avoiding forbidden pairs of arcs.
- The sets of (s,t)-TSP-paths with the above structure are of exponential size.
- The shortest (s,t)-TSP-path over each of the sets above can be determined in polynomial time.

Outlook

Outlook

- Get an improvement of the time complexity to $O(n^2)$ or $O(n^3)$ for the Path-TSP on Demidenko distance matrices

Outlook

- Get an improvement of the time complexity to $O(n^2)$ or $O(n^3)$ for the Path-TSP on Demidenko distance matrices
- Use the exponential neighborhoods above in local search algorithms for the Path-TSP by exploiting that the optimization over these neighbourhodds can be done in polynomial time

  (in the wein of Deineko and Woeginger 2000, Gutin, Yeo, Zverovich 2002, Orlin and Sharma 2004.

Outlook

- Get an improvement of the time complexity to $O(n^2)$ or $O(n^3)$ for the Path-TSP on Demidenko distance matrices
- Use the exponential neighborhoods above in local search algorithms for the Path-TSP by exploiting that the optimization over these neighbourhodds can be done in polynomial time

  (in the wein of Deineko and Woeginger 2000, Gutin, Yeo, Zverovich 2002, Orlin and Sharma 2004.

Thanks a lot for the attention!