Politechnika Wrocławska

Wybrane problemy odpornej optymalizacji dyskretnej

z możliwością modyfikacji

Autor: Tomasz Strzałka

Promotor: dr hab. Paweł Zieliński, prof. PWr

Wydział Podstawowych Problemów Techniki

3 lipca 2016

Plan prezentacji

Plan

- Problem: *Incremental Minimum Spanning Tree*,
  - idea,
  - LP a *Binary-Based IMST*.
- Problem: *Recoverable Robust Minimum Spanning Tree*,
  - problem: Min-Max,
  - problem adwersarza.
- Algorytm: *Tabu Search*.

Plan prezentacji

Plan

- Problem: *Incremental Minimum Spanning Tree*,
  - idea,
  - LP a *Binary-Based IMST*.
- Problem: *Recoverable Robust Minimum Spanning Tree*,
  - problem: Min-Max,
  - problem adwersarza.
- Algorytm: *Tabu Search*.

Plan prezentacji

Plan

- Problem: *Incremental Minimum Spanning Tree*,
  - idea,
  - LP a *Binary-Based IMST*.
- Problem: *Recoverable Robust Minimum Spanning Tree*,
  - problem: Min-Max,
  - problem adwersarza.
- Algorytm: *Tabu Search*.

Plan prezentacji

Plan

- Problem: *Incremental Minimum Spanning Tree*,
  - idea,
  - LP a *Binary-Based IMST*.
- Problem: *Recoverable Robust Minimum Spanning Tree*,
  - problem: Min-Max,
  - problem adwersarza.
- Algorytm: *Tabu Search*.

Plan prezentacji

Plan

- Problem: *Incremental Minimum Spanning Tree*,
  - idea,
  - LP a *Binary-Based IMST*.
- Problem: *Recoverable Robust Minimum Spanning Tree*,
  - problem: Min-Max,
  - problem adwersarza.
- Algorytm: *Tabu Search*.

Plan prezentacji

Plan

- Problem: *Incremental Minimum Spanning Tree*,
  - idea,
  - LP a *Binary-Based IMST*.
- Problem: *Recoverable Robust Minimum Spanning Tree*,
  - problem: Min-Max,
  - problem adwersarza.
- Algorytm: *Tabu Search*.

Plan prezentacji

Plan

- Problem: *Incremental Minimum Spanning Tree*,
  - idea,
  - LP a *Binary-Based IMST*.
- Problem: *Recoverable Robust Minimum Spanning Tree*,
  - problem: Min-Max,
  - problem adwersarza.
- Algorytm: *Tabu Search*.

Incremental Minimum Spanning Tree

Definicja

$$G = (V, E)\,, \quad |V| = n, \quad |E| = m,$$
$$\mathbf{s} = [c_1^{\mathbf{s}}, c_2^{\mathbf{s}}, \ldots, c_m^{\mathbf{s}}]\,,$$
$$G_{\mathbf{s}} = (V, E, \mathbf{s})\,,$$
$$\forall i \in \{1, 2, \ldots, m\} \quad c_{e_i} = c_i^{\mathbf{s}}.$$

$$\exists\, \mathbf{s}, \mathbf{s}' \quad : \quad \exists c_i^{\mathbf{s}} \neq c_i^{\mathbf{s}'}\,,$$
$$T_{\mathbf{s}}^* = \min arg_{T \in \mathcal{T}(G_{\mathbf{s}})} \sum_{e \in T} c_e\,.$$

Incremental Minimum Spanning Tree

Definicja

$$G = (V, E), \quad |V| = n, \quad |E| = m,$$
$$\mathbf{s} = [c_1^{\mathbf{s}}, c_2^{\mathbf{s}}, \ldots, c_m^{\mathbf{s}}],$$
$$G_{\mathbf{s}} = (V, E, \mathbf{s}),$$
$$\forall i \in \{1, 2, \ldots, m\} \quad c_{e_i} = c_i^{\mathbf{s}}.$$

$$\exists\, \mathbf{s}, \mathbf{s}' \quad : \quad \exists c_i^{\mathbf{s}} \neq c_i^{\mathbf{s}'},$$
$$T_{\mathbf{s}}^* = \min arg_{T \in \mathcal{T}(G_{\mathbf{s}})} \sum_{e \in T} c_e.$$

Incremental Minimum Spanning Tree

Definicja

$$G = (V, E), \quad |V| = n, \quad |E| = m,$$
$$\mathbf{s} = [c_1^{\mathbf{s}}, c_2^{\mathbf{s}}, \ldots, c_m^{\mathbf{s}}],$$
$$G_{\mathbf{s}} = (V, E, \mathbf{s}),$$
$$\forall i \in \{1, 2, \ldots, m\} \quad c_{e_i} = c_i^{\mathbf{s}}.$$

$$\exists\, \mathbf{s}, \mathbf{s}' \quad : \quad \exists c_i^{\mathbf{s}} \neq c_i^{\mathbf{s}'},$$
$$T_{\mathbf{s}}^* = \min arg_{T \in \mathcal{T}(G_{\mathbf{s}})} \sum_{e \in T} c_e.$$

Incremental Minimum Spanning Tree

Definicja

$$G = (V, E), \quad |V| = n, \quad |E| = m,$$
$$\mathbf{s} = [c_1^{\mathbf{s}}, c_2^{\mathbf{s}}, \ldots, c_m^{\mathbf{s}}],$$
$$G_{\mathbf{s}} = (V, E, \mathbf{s}),$$
$$\forall i \in \{1, 2, \ldots, m\} \quad c_{e_i} = c_i^{\mathbf{s}}.$$

$$\exists\, \mathbf{s}, \mathbf{s}' \quad : \quad \exists c_i^{\mathbf{s}} \neq c_i^{\mathbf{s}'},$$
$$T_{\mathbf{s}}^* = \min arg_{T \in \mathcal{T}(G_{\mathbf{s}})} \sum_{e \in T} c_e.$$

Incremental Minimum Spanning Tree

Definicja

$$G = (V, E), \quad |V| = n, \quad |E| = m,$$
$$\mathbf{s} = [c_1^{\mathbf{s}}, c_2^{\mathbf{s}}, \ldots, c_m^{\mathbf{s}}],$$
$$G_{\mathbf{s}} = (V, E, \mathbf{s}),$$
$$\forall i \in \{1, 2, \ldots, m\} \quad c_{e_i} = c_i^{\mathbf{s}}.$$

$$\exists\, \mathbf{s}, \mathbf{s}' \quad : \quad \exists c_i^{\mathbf{s}} \neq c_i^{\mathbf{s}'},$$
$$T_{\mathbf{s}}^* = \min arg_{T \in \mathcal{T}(G_{\mathbf{s}})} \sum_{e \in T} c_e.$$

Incremental Minimum Spanning Tree

Definicja

$$G = (V, E), \quad |V| = n, \quad |E| = m,$$
$$\mathbf{s} = [c_1^{\mathbf{s}}, c_2^{\mathbf{s}}, \ldots, c_m^{\mathbf{s}}],$$
$$G_{\mathbf{s}} = (V, E, \mathbf{s}),$$
$$\forall i \in \{1, 2, \ldots, m\} \quad c_{e_i} = c_i^{\mathbf{s}}.$$

$$\exists\, \mathbf{s}, \mathbf{s}' \quad : \quad \exists c_i^{\mathbf{s}} \neq c_i^{\mathbf{s}'},$$
$$T_{\mathbf{s}}^* = \min arg_{T \in \mathcal{T}(G_{\mathbf{s}})} \sum_{e \in T} c_e.$$

Incremental Minimum Spanning Tree

Definicja c.d.

$$T_{\mathbf{s}}^{*} = \min arg_{T \in \mathcal{T}(G)} \sum_{e \in T} c_e^{\mathbf{s}},$$

$$f : \mathcal{T}(G) \times \mathcal{T}(G) \rightarrow \mathbb{Z}^{+},$$

$$f(T, T') = |T \setminus T'| = |T' \setminus T|,$$

$$\mathcal{T}(G, T, k) = \{T' : T' \in \mathcal{T}(G) \wedge f(T, T') \leqslant k\},$$

$$T^{*} = \min arg_{T' \in \mathcal{T}(G,T,k)} \sum_{e \in T'} c_e.$$

Incremental Minimum Spanning Tree

Definicja c.d.

$$T_{\mathbf{s}}^{*} = \min arg_{T \in \mathcal{T}(G)} \sum_{e \in T} c_e^{\mathbf{s}},$$

$$f : \mathcal{T}(G) \times \mathcal{T}(G) \to \mathbb{Z}^{+},$$

$$f(T, T') = |T \setminus T'| = |T' \setminus T|,$$

$$\mathcal{T}(G, T, k) = \{T' : T' \in \mathcal{T}(G) \wedge f(T, T') \leqslant k\},$$

$$T^{*} = \min arg_{T' \in \mathcal{T}(G,T,k)} \sum_{e \in T'} c_e.$$

Incremental Minimum Spanning Tree

Definicja c.d.

$$T_{\mathbf{s}}^{*} = \min arg_{T \in \mathcal{T}(G)} \sum_{e \in T} c_e^{\mathbf{s}},$$

$$f : \mathcal{T}(G) \times \mathcal{T}(G) \to \mathbb{Z}^{+},$$

$$f(T, T') = |T \setminus T'| = |T' \setminus T|,$$

$$\mathcal{T}(G, T, k) = \{T' : T' \in \mathcal{T}(G) \wedge f(T, T') \leqslant k\},$$

$$T^{*} = \min arg_{T' \in \mathcal{T}(G,T,k)} \sum_{e \in T'} c_e.$$

Incremental Minimum Spanning Tree

Definicja c.d.

$$T_{\mathbf{s}}^{*} = \min arg_{T \in \mathcal{T}(G)} \sum_{e \in T} c_e^{\mathbf{s}},$$

$$f : \mathcal{T}(G) \times \mathcal{T}(G) \to \mathbb{Z}^{+},$$

$$f(T, T') = |T \setminus T'| = |T' \setminus T|,$$

$$\mathcal{T}(G, T, k) = \{T' : T' \in \mathcal{T}(G) \ \wedge \ f(T, T') \leqslant k\},$$

$$T^{*} = \min arg_{T' \in \mathcal{T}(G,T,k)} \sum_{e \in T'} c_e.$$

Problem Incremental - LP

Model LP

$$
\begin{aligned}
\min \quad & \sum_{e \in E} c_e \cdot x_e, \\
\text{s.t.} \quad & A \cdot \mathbf{x} = \mathbf{b}, \\
& \sum_{e \in T^* \setminus T^*_{\mathbf{s}}} x_e \leqslant k, \\
& x_e \geqslant 0, \ \forall e \in E,
\end{aligned}
$$

Problem Incremental - relaksacja

Relaksacja ograniczeń modelu

$$\min \quad \sum_{e \in E} c_e \cdot x_e + \lambda \cdot \left(\sum_{e \in T^* \setminus T_s^*} x_e - k \right),$$

$$\text{s.t.} \quad A \cdot \mathbf{x} = \mathbf{b},$$

$$x_e \geqslant 0, \qquad \forall e \in E,$$

Problem Incremental - cel

Definicja c.d.

$$\sum_{e_i \in E \setminus T_s^*} c_i \cdot x_i + \sum_{e_i \in T_s^*} (c_i - \lambda) \cdot x_i,$$

$$f\left(T^*, T_s^*\right) \leqslant k \ \wedge \ \lambda = 0 \quad \text{lub}$$
$$f\left(T^*, T_s^*\right) = k \ \wedge \ \lambda \neq 0,$$

$$\lambda^* \ : \ T' \in \mathcal{T}\left(G_{s(\lambda^*)}, T, k\right) \ \wedge \ f\left(T, T'\right) = k,$$

Problem Incremental - cel

Definicja c.d.

$$\sum_{e_i \in E \setminus T_{\mathbf{s}}^*} c_i \cdot x_i + \sum_{e_i \in T_{\mathbf{s}}^*} (c_i - \lambda) \cdot x_i,$$

$$f\left(T^*, T_{\mathbf{s}}^*\right) \leqslant k \ \wedge \ \lambda = 0 \quad \text{lub}$$
$$f\left(T^*, T_{\mathbf{s}}^*\right) = k \ \wedge \ \lambda \neq 0,$$

$$\lambda^* \ : \ T' \in \mathcal{T}\left(G_{\mathbf{s}(\lambda^*)}, T, k\right) \ \wedge \ f\left(T, T'\right) = k,$$

Problem Incremental - cel

Definicja c.d.

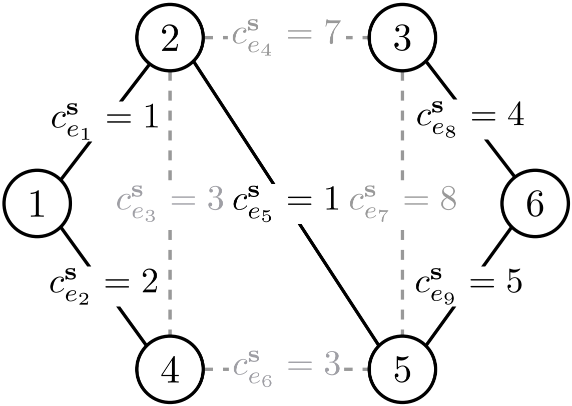
$$\sum_{e_i \in E \setminus T_\mathbf{s}^*} c_i \cdot x_i + \sum_{e_i \in T_\mathbf{s}^*} (c_i - \lambda) \cdot x_i,$$

$$f\left(T^*, T_\mathbf{s}^*\right) \leqslant k \ \wedge \ \lambda = 0 \quad \text{lub}$$
$$f\left(T^*, T_\mathbf{s}^*\right) = k \ \wedge \ \lambda \neq 0,$$

$$\lambda^* \ : \ T' \in \mathcal{T}\left(G_{\mathbf{s}(\lambda^*)}, T, k\right) \ \wedge \ f\left(T, T'\right) = k,$$

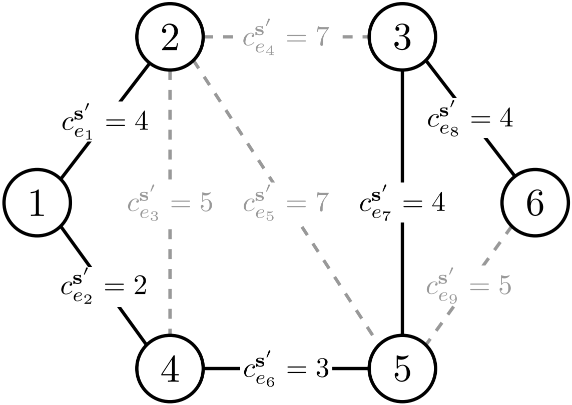
Problem Incremental - przykłady

Scenariusz **s**, $k = 1$, $T^*_{\mathbf{s}}$



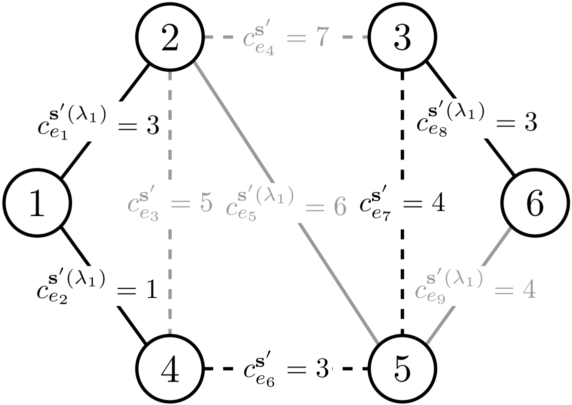
Problem Incremental - przykłady

Scenariusz $\mathbf{s}'$, $k = 1$, $T^*_{\mathbf{s}'}$



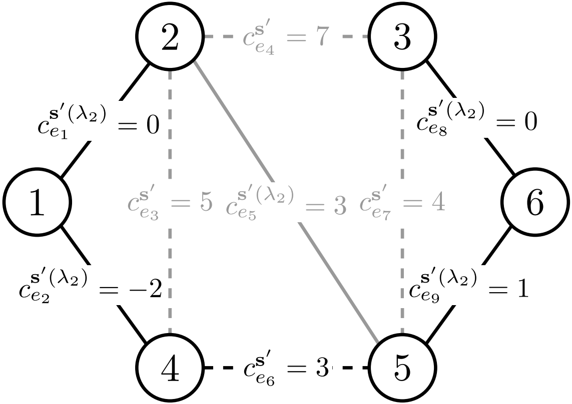
Problem Incremental - przykłady

Scenariusz $\mathbf{s}'_{\lambda_1}$, $k = 1$, $T(\lambda_1)$



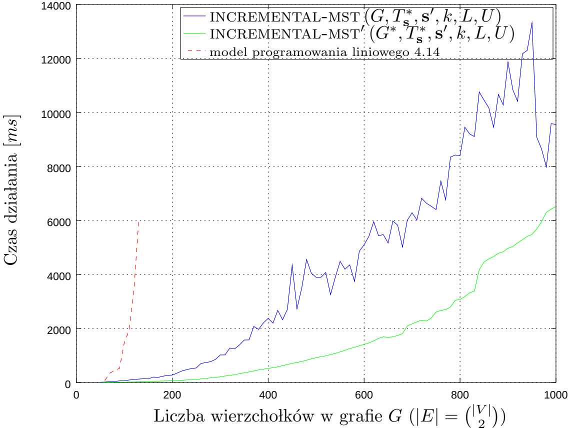
Problem Incremental - przykłady

Scenariusz $\mathbf{s}'_{\lambda_2}$, $k = 1$, $T(\lambda_2)$



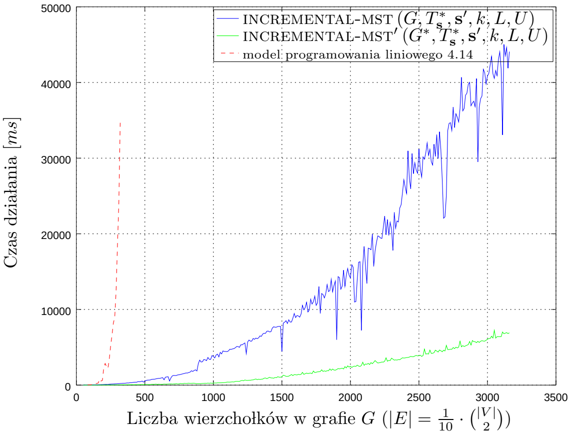
Problem Incremental - czas działania

Algorytmy iteracyjne a model LP



Problem Incremental - czas działania

Algorytmy iteracyjne a model LP



Problem Recoverable - idea

Min-Max

$$\min_{\mathbf{x}\in X} \max_{\mathbf{s}\in S} v\left(\mathbf{x},\mathbf{s}\right)$$

$$\mathbf{s} \in S = \left\{\mathbf{s}^1, \mathbf{s}^2, \ldots, \mathbf{s}^n\right\}$$

Recoverable Robust

$$\min_{\mathbf{x}\in X} \left(v\left(\mathbf{x},\mathbf{s}\right) + \max_{\mathbf{s}'\in S} \min_{\mathbf{y}\in S_{\mathbf{x}}} v\left(\mathbf{y},\mathbf{s}'\right) \right)$$

Problem Recoverable dla IMST - idea

Problem adwersarza

$$\max_{\mathbf{s}'\in S}\min_{\mathbf{y}\in S_{\mathbf{x}}} v\left(\mathbf{y},\mathbf{s}'\right)$$

Recoverable Robust Incr. Minimum Spanning Tree

$$\min_{T\in\mathcal{T}(G)}\left(\sum_{e\in T} c_e^{\mathbf{s}} + \max_{\mathbf{s}'\in S}\ \min_{T'\in\mathcal{T}(G_{\mathbf{s}'},T,k)}\ \sum_{e'\in T'} c_{e'}^{\mathbf{s}'}\right)$$

Tabu Search

Recoverable Robust Incr. Minimum Spanning Tree

Problem NP-trudny, nieaproksymowalny.
Dowód poprzez sprowadzenie problemu decyzyjnego Minimum Degree Spanning Tree (NP-trudny) do RRIMST.

Adam Kasperski, Adam Kurpisz, and Paweł Zieliński. Recoverable Robust Combinatorial Optimization Problems, *strony* 147-–153. *Springer International Publishing, Cham, 2014.*

Tabu Search - idea

Kroki

- Weź losowy punkt z przestrzeni dopuszczalnych rozwiązań,
- wykonaj **ruch** w **otoczeniu** obecnego rozwiązania na podstawie **funkcji oceny ruchu**,
- powtarzaj aż nie zajdzie pewien warunek,
- wybierz inne losowe rozwiązanie, dalekie od tego, wybranego na początku,
- powtarzaj powyższe kroki aż do momentu, gdy zajdzie pewien warunek końca.

Tabu Search - idea

Kroki

- Weź losowy punkt z przestrzeni dopuszczalnych rozwiązań,
- wykonaj **ruch** w **otoczeniu** obecnego rozwiązania na podstawie **funkcji oceny ruchu**,
- powtarzaj aż nie zajdzie pewien warunek,
- wybierz inne losowe rozwiązanie, dalekie od tego, wybranego na początku,
- powtarzaj powyższe kroki aż do momentu, gdy zajdzie pewien warunek końca.

Tabu Search - idea

Kroki

- Weź losowy punkt z przestrzeni dopuszczalnych rozwiązań,
- wykonaj **ruch** w **otoczeniu** obecnego rozwiązania na podstawie **funkcji oceny ruchu**,
- powtarzaj aż nie zajdzie pewien warunek,
- wybierz inne losowe rozwiązanie, dalekie od tego, wybranego na początku,
- powtarzaj powyższe kroki aż do momentu, gdy zajdzie pewien warunek końca.

Tabu Search - idea

Kroki

- Weź losowy punkt z przestrzeni dopuszczalnych rozwiązań,
- wykonaj **ruch** w **otoczeniu** obecnego rozwiązania na podstawie **funkcji oceny ruchu**,
- powtarzaj aż nie zajdzie pewien warunek,
- wybierz inne losowe rozwiązanie, dalekie od tego, wybranego na początku,
- powtarzaj powyższe kroki aż do momentu, gdy zajdzie pewien warunek końca.

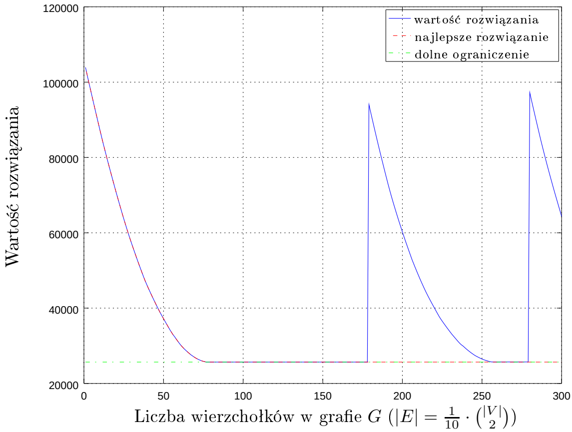
Tabu Search - idea

Kroki

- Weź losowy punkt z przestrzeni dopuszczalnych rozwiązań,
- wykonaj **ruch** w **otoczeniu** obecnego rozwiązania na podstawie **funkcji oceny ruchu**,
- powtarzaj aż nie zajdzie pewien warunek,
- wybierz inne losowe rozwiązanie, dalekie od tego, wybranego na początku,
- powtarzaj powyższe kroki aż do momentu, gdy zajdzie pewien warunek końca.

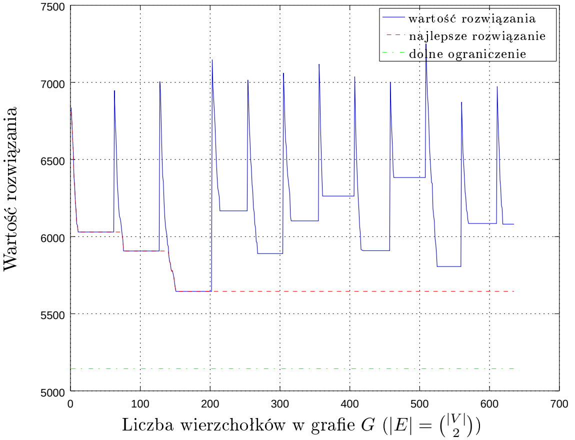
Tabu Search - wyniki

Funkcja oceny ruchu - różnica wartości rozwiązań



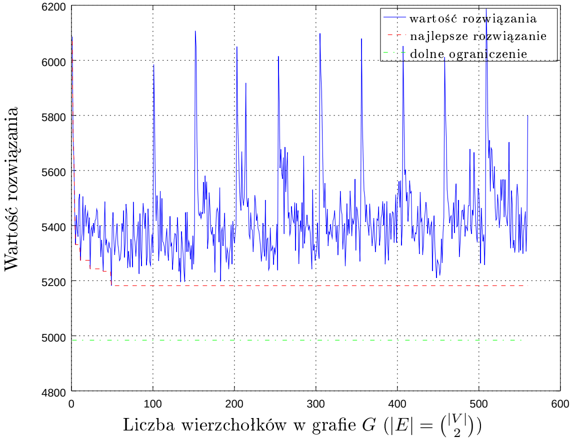
Tabu Search - wyniki

$k = 1$

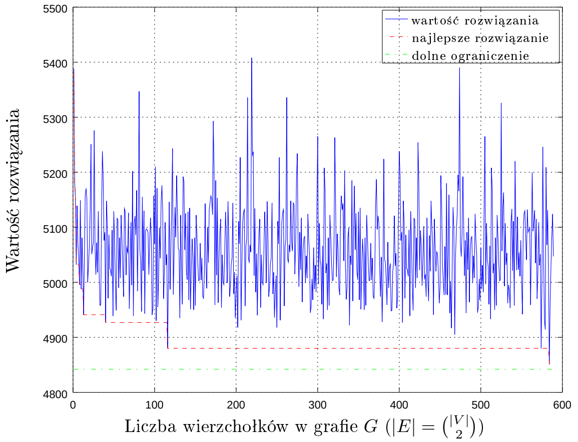


Tabu Search - wyniki

$k = 7$



Tabu Search - wyniki



Tabu Search - wyniki

#	$v_{\mathbf{ts}}(T,S)$	$\frac{v_{\mathbf{ts}}(T,S)-v_{\mathbf{ts}}^{LB*}}{v_{\mathbf{ts}}^{LB*}}$
1	5388	$11,28\%$
3	5151	$6,38\%$
8	4996	$3,18\%$
10	4988	$3,02\%$
13	4941	$2,04\%$
40	4927	$1,76\%$
116	4880	$0,78\%$
584	4851	$0,19\%$

Tabu Search - parametry

Funkcje oceny ruchu

$$Mval\,(T, T_1) = v_{\mathrm{RRIMST}}\,(T, S) - v_{\mathrm{RRIMST}}\,(T_1, S)\,,$$

$$Mval\,(T, T_1) = \alpha_1 \cdot (v_{\mathrm{RRIMST}}\,(T, S) - v_{\mathrm{RRIMST}}\,(T_1, S)) +$$
$$+\alpha_2 \cdot \frac{\mathrm{R}\,[i, j]}{it} + \alpha_3 \cdot \frac{\mathrm{R}\,[k, l]}{it} + \alpha_4 \cdot \mathrm{MR}\,[i, j] + \alpha_5 \cdot \mathrm{MR}\,[k, l]$$

Tabu Search - parametry

Funkcje oceny ruchu

$$Mval\,(T, T_1) = v_{\mathrm{RRIMST}}\,(T, S) - v_{\mathrm{RRIMST}}\,(T_1, S)\,,$$

$$Mval\,(T, T_1) = \alpha_1 \cdot (v_{\mathrm{RRIMST}}\,(T, S) - v_{\mathrm{RRIMST}}\,(T_1, S)) +$$

$$+\alpha_2 \cdot \frac{\mathrm{R}\,[i,j]}{it} + \alpha_3 \cdot \frac{\mathrm{R}\,[k,l]}{it} + \alpha_4 \cdot \mathrm{MR}\,[i,j] + \alpha_5 \cdot \mathrm{MR}\,[k,l]$$

Tabu Search - parametry

Dolne ograniczenie

$$|S| = 1 \rightarrow \min_{\mathbf{x}\in X} v(\mathbf{x},\mathbf{s}) + \max_{\mathbf{s}'\in S} \min_{\mathbf{y}\in X_{\mathbf{x}}^{k}} v(\mathbf{y},\mathbf{s}') \equiv$$

$$\equiv \min_{\mathbf{x}\in X} v(\mathbf{x},\mathbf{s}) + \min_{\mathbf{y}\in X_{\mathbf{x}}^{k}} v(\mathbf{y},\mathbf{s}'),$$

$$\mathrm{LB} \leqslant \mathrm{LB}^{*} = \max_{\mathrm{LB}\in\mathcal{LB}} \mathrm{LB} \leqslant v^{*}_{\mathrm{RRIMST}}.$$