

Некоторые методы решения маршрутных задач с условиями предшествования

05.13.18 — Мат. моделирование, численные методы и комплексы программ
Диссертация на соискание ученой степени кандидата физ.-мат. наук

Ярослав Салий^{ab} yvs314@gmail.com

Научный руководитель:
Александр Георгиевич Ченцов
д.ф.-м.н., член-корр. РАН

Научный семинар по информационным технологиям ЮУрГУ
Челябинск, 5 декабря, 2017

^aИнститут математики и механики им. Н.Н. Красовского УрО РАН, *Отдел управляемых систем*

^bУральский федеральный университет, *Кафедра прикладной математики и механики*

Цель: Разработка методов и алгоритмов решения задач **маршрутизации** перемещений **с ограничениями** и функциями стоимости, допускающими зависимость от списка заданий

- Задача 1** Построение конструкций решения на основе широко понимаемого **динамического программирования**
- Задача 2** Оценка влияния **условий предшествования** на вычислительную сложность
- Задача 3** Разработка и реализация алгоритмов и схем их параллелизации, проведение вычислительных экспериментов

TSP-PC. Постановка задачи

$$\mathfrak{C}[\alpha] \xrightarrow{\alpha \in \mathbb{A}} \min.$$

(TSP-PC)

- $\overline{1, n}$: города/их индексы
 - 0: база, фиксированная точка *старта*;
 $\mathfrak{t} = n + 1$: терминал, фиксированная точка *окончания*
 - $\mathfrak{c}: \overline{0, n} \times \overline{1, n + 1} \rightarrow \mathbb{R}$: стоимость перемещений
 - $\mathfrak{C}[\alpha] = \mathfrak{c}(0, \alpha_1) \oplus \dots \oplus \mathfrak{c}(\alpha_n, \mathfrak{t})$: суммарная стоимость маршрута
-
- $P = (\overline{1, n}, <_P)$: частичный порядок^a; условия предшествования
 - допустимые маршруты, *линейные продолжения* P :

$$\mathbb{A} \triangleq \left\{ \alpha \in (\text{bi})[\overline{1, n}] \mid \forall a, b \in \overline{1, n} (a <_P b) \rightarrow (\alpha^{-1}(a) < \alpha^{-1}(b)) \right\}$$

^aеще бывает *орграф без циклов* и *фундированное отношение* — эквивалентно

Как ни называйте судно

TSP-PC: TSP with Precedence Constraints (Lawler, 1979);
(Bianco, Mingozzi, Ricciardelli, Spadoni, 1992)

Задача курьера (Плотинский, 1973); (Меламад, Сергеев, Сигал, 1988)

SOP: Sequential Ordering Problem (Escudero, 1988)

PCTSP^a, PCATS: Precedence Constrained (Asymmetric) TSP
(Ascheuer, Junger, Reinelt, 2000)

все равно придется посетить каждый город *единственный* раз, соблюдая условия предшествования, за наименьшую цену

^a никак не связано с PCTSP, Prize Collecting Traveling Salesman Problem

SD-BGTSP-PC. «Геометрия»

- непустое *объемлющее* множество X
- база $x^0 \in X$
- число мегаполисов n , $n \in \mathbb{N}$, $n \geq 2$
- сами мегаполисы $M_1 \in \text{Fin } X, \dots, M_n \in \text{Fin } X$
- ограничения на вход-выход $\mathbb{M}_1 \in \mathcal{P}'(M_1 \times M_1), \dots, \mathbb{M}_n \in \mathcal{P}'(M_n \times M_n)$;
УП $z \in \mathbb{M}_j$ описывает способ пройти через M_j :
 - $\text{pr}_1(z)$ город-вход
 - $\text{pr}_2(z)$ город-выход

Мегаполисы *дизъюнкты*, база — отдельно:

$$(x^0 \notin M_j \ \forall j \in \overline{1, n}) \& (M_p \cap M_q = \emptyset \ \forall p \in \overline{1, n} \ \forall q \in \overline{1, n} \setminus \{p\})$$

SD-BGTSP-PC. Еще чуть геометрии

$$\mathbb{X} \triangleq \{x^0\} \cup \left(\bigcup_{i=1}^n M_i \right) \text{ Все города}$$

$$\mathbb{M}_j^{(\text{in})} \triangleq \{\text{pr}_1(z) : z \in \mathbb{M}_j\} \forall j \in \overline{1, n} \text{ Все возможные } \textit{входы} \text{ в } j\text{-й мегаполис}$$

$$\mathbb{X}_{\text{in}} \triangleq \{\emptyset\} \cup \left(\bigcup_{i=1}^n \mathbb{M}_i^{(\text{in})} \right) \text{ Все города-входы}$$

$$\mathbb{M}_j^{(\text{out})} \triangleq \{\text{pr}_2(z) : z \in \mathbb{M}_j\} \forall j \in \overline{1, n} \text{ Все возможные } \textit{выходы} \text{ из } j\text{-го мегаполиса}$$

$$\mathbb{X}_{\text{out}} \triangleq \{x^0\} \cup \left(\bigcup_{i=1}^n \mathbb{M}_i^{(\text{out})} \right) \text{ Все города-выходы}$$

SD-BGTSP-PC. Критерий качества

Множество всех трасс, *согласующихся* с маршрутом α :

$$\mathcal{Z}_\alpha \triangleq \left\{ (z_i)_{i=0}^n \in \mathbb{Z} \mid \left(z_0 = (\emptyset, x^0) \right) \& \left(z_t \in \mathbb{M}_{\alpha(t)} \ \forall t \in \overline{1, n} \right) \right\}.$$

Стоимость обхода по **трассе** $(z_i)_{i=0}^n \in \mathcal{Z}_\alpha$, согласующейся с **маршрутом** $\alpha \in \mathbb{A}$ составляет

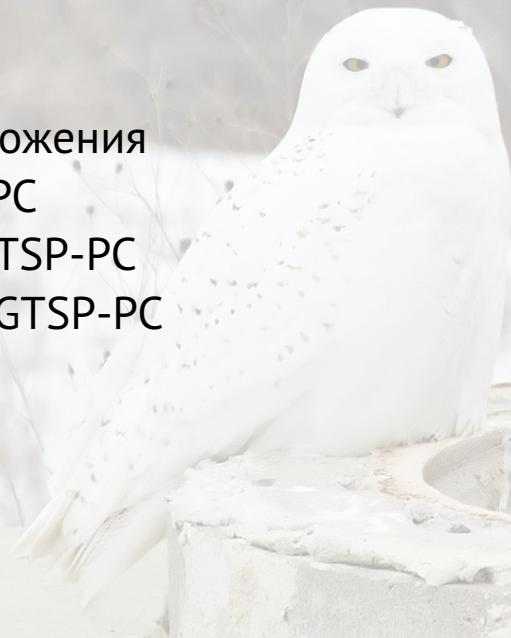
$$\begin{aligned} \mathfrak{C}_\alpha [(z_i)_{i=0}^n] \triangleq \max_{t \in \overline{0, n-1}} & \left[\mathfrak{C}(\text{pr}_2(z_t), \text{pr}_1(z_{t+1}), \{\alpha(s) : s \in \overline{t+1, n}\}) \right. \\ & \left. + c_{\alpha(t+1)}(z_{t+1}, \{\alpha(s) : s \in \overline{t+1, n}\}) \right] \end{aligned}$$

$\mathfrak{N} \triangleq \mathcal{P}(\overline{1, n})$ все списки заданий

$\mathfrak{C} \in \mathcal{R}_+(\mathbb{X}_{\text{out}} \times \mathbb{X}_{\text{in}} \times \mathfrak{N})$ стоимость **внешних перемещений**

$c_i \in \mathcal{R}_+(\mathbb{X}_{\text{in}} \times \mathbb{X}_{\text{out}} \times \mathfrak{N})$, $i \in \overline{1, n}$ стоимость **внутренних работ**

Приложения
TSP-PC
SD-GTSP-PC
SD-BGTSP-PC



Вертолет и нефтяные платформы



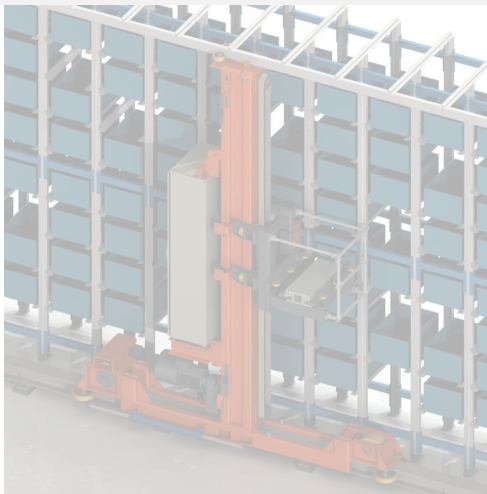
(Fiala Timlin, Pulleyblank, 1992): 45 нефтяных платформ в Нигерии, по заказу Mobil



Оптимизация работы крана-штабелера

Автоматизированные системы хранения

(Ascheuer, 1995)
Для завода Siemens Nixdorf
— 30% холостого хода



Маршрутизация инструмента машин листовой резки

Sequence Dependent Generalized TSP-PC

(Castelino, D'Souza, Wright, 2003);
(Dewil, Vansteenwegen, Catrysse, 2015);
(Chentsov, Petunin, 2016);

- (Past) Sequence Dependent^a: жесткость, тепловое расширение
- Generalized: возможны разные точки *врезки*

Формально: минимизация *холостого* хода

Особенность: скорее **constraint satisfaction** чем оптимизация

^aтермин из (Koulamas, Kyriaris, 2008), задача о расписании с зависимостью времени переналадки от предыстории

Робот-инспектор

SD-BGTSP-PC



L.Gouveia, M.Ruthmair, D.Santos, *Routing Problems for Electric Vehicles with Load-Dependent Energy Consumption*, ICCL'16, 7th International Conference on Computational Logistics (September 07–09, 2016, Lisbon, Portugal). 2016. PP. 9–10
EVRP – задача развозки для электромобилей

Bottleneck: на узкие места
Опт. емкость аккумулятора



Chentsov A. G., Salii Y. V. A model of “nonadditive” routing problem where the costs depend on the set of pending tasks // Vestnik YuUrGU. Ser. Mat. Model. Progr. 2015. Vol. 8, no. 1. PP. 24–45.

Как решают (TSP-PC)?

- MILP Mixed Integer-Linear Programming (Balas, Fischetti, Pulleyblank, 1993); (Ascheuer, 1995+);
(Gouveia, Ruthmair, 2015) — **лучшие результаты**^a
- DP Динамическое программирование
(Schrage, Baker, 1978); (Lawler, 1979);
(Bianco, Mingozzi, Ricciardelli, Spadoni, 1992);
(Ченцов, Ченцов, 2004+)
- B&B Метод ветвей и границ
(Многие Авторы, 1985+)
- MDD Multivalued Decision Diagrams
(Cire, van Hoeve, 2013)

^aза исключением (Salii, 2017) — впервые оптимально решена **ry48p.3.sop**, через ДП

Как решают SD-TSP-PC?

Функция стоимости перемещений зависит от предыстории

CP/MDD Constraint Programming with Multivalued Decision Diagrams

(Kinable, Cire, van Hoeve, 2016);

зависимость от позиции («TD-SOP»), задачи на основе TSPLIB, не более 98 городов

DP Динамическое программирование

(Сесекин, Ченцов, Ченцов, 2010) – *впервые* поставлена SD-GTSP-PC (Salii, 2017):

- +7 закрытых задач, включая `rbg253a.sop` – 253 города
- улучшены все оценки сверху для еще не закрытых, через *усеченное* ДП

Зависимость типа *traveling deliveryman*

$\vartheta(i, j)$ – расстояние от i до j ; $c: \mathcal{P}(\overline{1, n}) \times \overline{0, n} \times \overline{1, n+1} \rightarrow \mathbb{R}$

$$c_{\text{TD-SOP}}(K, i, j) \triangleq \begin{cases} (n+1)\vartheta(i, j) & (K = \emptyset) \wedge (i = 0); \\ (n - |K|)\vartheta(i, j) & (K \in \mathcal{P}'(\overline{1, n})) \wedge (i \neq 0). \end{cases}$$

Наборы тестовых экземпляров TSP-PC (SOP)

- **TSPLIB** 7–380 городов; (Reinelt, 1991); (Ascheuer, Escudero, 1995+)
- **SOPLIB** 100–700 городов; (Montemanni, Smith, Gambardella, 2008)
- **COMPILERS** до 600 городов??? (Shobaki, Jamal, 2015)

Характерная размерность точного решения

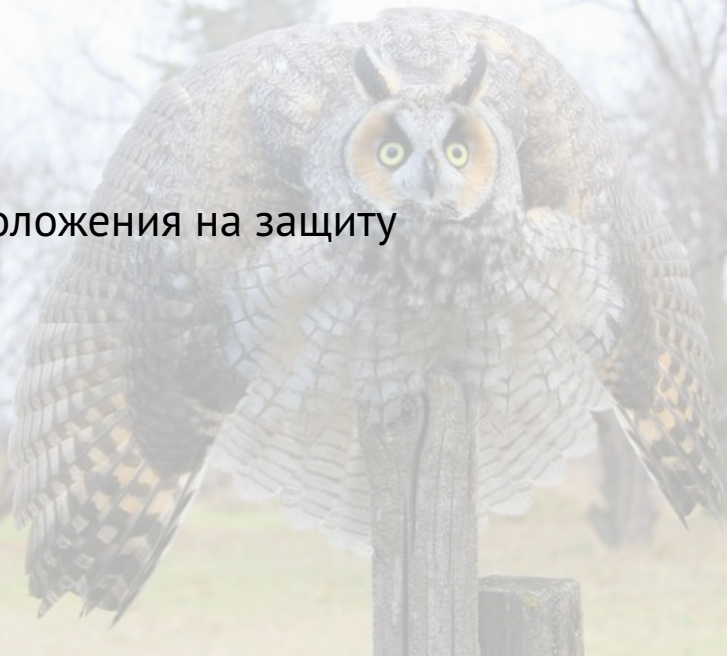
Обычная TSP

~ 80,000 городов (симм.)
~ 20,000 городов (асимм.)

TSP-PC

48 городов **нет точного решения**
(TSPLIB:ry48p2.sop)

Положения на защиту



I. Априорная оценка применимости ДП для TSP-PC

Максима: «Хватит памяти — вычислить недолго^а»

- 1 Оценки количества состояний ДП сверху и снизу
 - 2 Оценка временной сложности ДП
 - 3 Расчет оценок для экземпляров TSP-PC из TSPLIB
-
- 4 Критерий **практической** разрешимости экземпляра задачи через ДП



Salii Y. V. Order-theoretic characteristics and dynamic programming for Precedence Constrained Traveling Salesman Problem // Proceedings of the Fourth Russian Finnish Symposium on Discrete Mathematics / Ed. by J. Karhumäki, Y. Matiyasevich, A. Saarela.

Vol. 26 of *TUCS Lecture Notes*.

Turku Centre for Computer Science, 2017.

P. 152–164. <http://urn.fi/URN:ISBN:978-952-12-3547-4>.

^авременная сложность *полиномиально* зависит от пространственной

II. Точное ДП для TSP-PC и TD-TSP-PC на задачах TSPLIB

ДП в сравнении с B&B, B&C (MILP), CP/MDD

Измерялось время счета (физическое), расход оперативной памяти

TSP-PC впервые закрыта ry48p.3.sop;

TD-TSP-PC впервые закрыты 7 задач:

ft53.4.sop, ft70.4.sop, rbg109a.sop, rbg150a.sop,
rbg174b.sop, rbg253a.sop, ry48p.3.sop



Salii Y. V. Revisiting Dynamic Programming for Precedence-Constrained Traveling Salesman Problem and Its Time-Dependent Generalization.

Revised 16.10.2017, under review in European Journal of Operational Research;
preprint available at Researchgate,

https:

[//www.researchgate.net/profile/Yaroslav_Salii/publications](https://www.researchgate.net/profile/Yaroslav_Salii/publications)

III. Эвристика «усеченное ДП» для TSP-PC и TD-TSP-PC на задачах TSPLIB

по сравнению с MILP и CP/MDD

TSP-PC

- При параметре $H = 100$, время счета около 1 сек, при этом для половины задач получены решения не более чем на 10% худшие оптимальных (наилучших известных)
- При параметре $H = 10^5$, только для 2 экземпляров (прямая процедура), либо только для 5 (попятная) значения на 10% или более хуже известных

TD-TSP-PC/TD-SOP

- Прямой вариант эвристики на $H = 10^5$ достигает **все** известные оптимальные решения кроме задачи p43.1.sop (на 3% хуже). **Улучшает** все оценки для еще не закрытых экземпляров из (Kinable, Cire, van Hoeve, 2016).
- Попятный вариант эвристики не слишком эффективен. Например, для p43.1.sop при $H = 10^5$ значение в 8 раз хуже оптимального.



Salii Y. V. Revisiting Dynamic Programming for Precedence-Constrained Traveling Salesman Problem and Its Time-Dependent Generalization.

Revised 16.10.2017, under review in European Journal of Operational Research; preprint available at Researchgate, [https:](https://www.researchgate.net/profile/Yaroslav_Salii/publications)

[//www.researchgate.net/profile/Yaroslav_Salii/publications](https://www.researchgate.net/profile/Yaroslav_Salii/publications)

IV. SD-BGTSP-PC

- Рассмотрена «богатая» задача (обобщенная, на узкие места, с условиями предшествования, с зависимостью от предыстории)
- Предложен и исследован метод *параллелизации* для систем с общей памятью; метод демонстрирует близкое к *линейному* масштабирование. Рассматривались задачи на 27 и 30 мегаполисов.
- Испытана эвристика усеченного ДП (допускающая предельный переход к точному методу). Получена оценка сложности для этой эвристики.



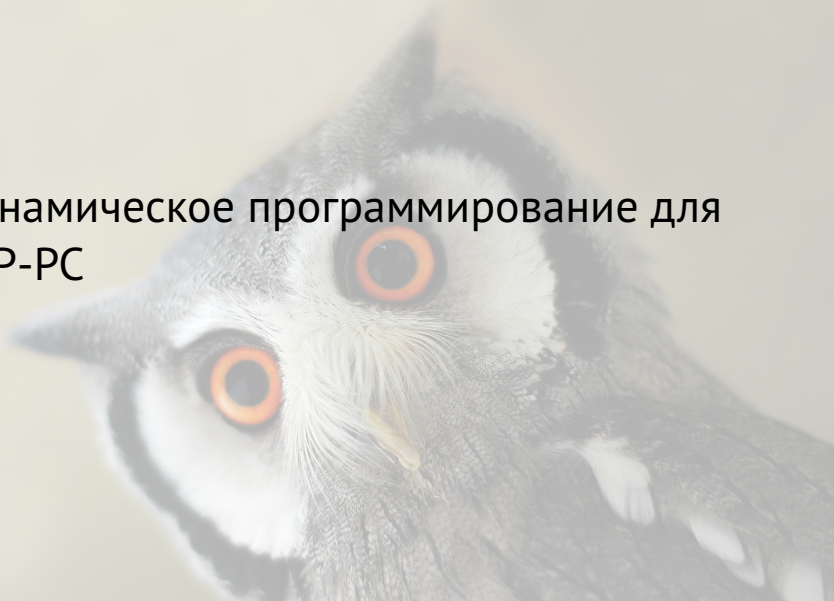
Chentsov A. G., Salii Y. V. A model of “nonadditive” routing problem where the costs depend on the set of pending tasks // Vestnik YuUrGU. Ser. Mat. Model. Progr. 2015. Vol. 8, no. 1. PP. 24–45.



Salii Y. V. Restricted Dynamic Programming Heuristic for Precedence Constrained Bottleneck Generalized TSP // Proceedings of the 1st Ural Workshop on Parallel, Distributed, and Cloud Computing for Young Scientists / Ed. by A. Sozykin, E. Akimova, D. Ustalov.

Vol. 1513. CEUR Workshop Proceedings, 2015. P. 85–108.
<http://ceur-ws.org/Vol-1513/#paper-10>.

Динамическое программирование для TSP-PC



Суть метода

Разбить задачу

$$\underbrace{0 \longrightarrow \overline{1, n} \longrightarrow \mathfrak{t}}_{\mathfrak{C}[\cdot] \rightarrow \min}$$

на взаимосвязанные *подзадачи* (состояния ДП)

прямое ДП: (K, x)

$$K \subset \overline{1, n}, x \in \overline{1, n+1}$$

$$\underbrace{0 \longrightarrow K \longrightarrow x \dots \mathfrak{t}}_{\mathfrak{C}[\cdot] \rightarrow \min}$$

(Held, Karp, 1962)

попятное ДП: (x, K)

$$K \subset \overline{1, n}, x \in \overline{0, n}$$

$$\underbrace{0 \dots x \longrightarrow K \longrightarrow \mathfrak{t}}_{\mathfrak{C}[\cdot] \rightarrow \min}$$

(Bellman, 1962)

Теперь — с условиями предшествования

Как *избавиться* от кучи^а работы без потери качества

Разбить задачу

$$0 \longrightarrow \overline{1, n} \longrightarrow \mathfrak{t}$$

$$\underbrace{\hspace{1.5cm}}_{\mathfrak{C}[\cdot] \rightarrow \min}$$

на взаимосвязанные *подзадачи* (состояния ДП)

прямое ДП: (K, x) , порядковые **идеалы**

$$\forall i \in K \forall j \in \overline{1, n} (j <_P i) \rightarrow (j \in K);$$

$$x \in \overline{1, n+1}$$

$$0 \longrightarrow K \longrightarrow x \dots \mathfrak{t}$$

$$\underbrace{\hspace{1.5cm}}_{\mathfrak{C}[\cdot] \rightarrow \min}$$

(Lawler, 1979)

попятное ДП: (x, K) , порядковые **фильтры**

$$\forall i \in K \forall j \in \overline{1, n} (j >_P i) \rightarrow (j \in K);$$

$$x \in \overline{0, n}$$

$$0 \dots x \longrightarrow K \longrightarrow \mathfrak{t}$$

$$\underbrace{\hspace{1.5cm}}_{\mathfrak{C}[\cdot] \rightarrow \min}$$

(Ченцов, Ченцов, 2004)

^а не является публичной офертой, может быть изменено без письменного уведомления в зависимости от частичного порядка P

Обход пространства состояний: Прямое ДП, в порядке мощности^a

Определение порядковых идеалов:

$$\mathcal{I} \triangleq \left\{ I \in \mathcal{P}(\overline{1, n}) \mid \forall i \in I \forall j \in \overline{1, n} (j <_p i) \rightarrow (j \in I) \right\}$$

вверх: $\mathbf{E}[I]: \mathcal{I} \rightarrow \mathcal{P}'(\overline{1, n})$

$$\begin{aligned} \mathbf{E}[I] &\triangleq \{m \in \overline{1, n} \setminus I \mid I \cup \{m\} \in \mathcal{I}\} \\ &= \text{Min} \{ \overline{1, n} \setminus I \} \end{aligned}$$

$$\begin{aligned} \text{Min}\{K\} &\triangleq \{m \in K \mid \forall k \in K \\ &\quad (m \leqslant_p k) \vee (m \parallel k)\} \end{aligned}$$

вниз: $\mathbf{I}[I]: \mathcal{I} \rightarrow \mathcal{P}'(\overline{1, n})$

$$\begin{aligned} \mathbf{I}[I] &\triangleq \{m \in I \mid I \setminus \{m\} \in \mathcal{I}\} \\ &= \text{Max}\{I\} \end{aligned}$$

$$\begin{aligned} \text{Max}\{K\} &\triangleq \{m \in K \mid \forall k \in K \\ &\quad (m \geqslant_p k) \vee (m \parallel k)\} \end{aligned}$$

^aпочетное упоминание: (Schrage, Baker, 1978), где обход близок к ПВГ по графу порядка

Поиск идеалов

$$\mathcal{I} \triangleq \left\{ I \in \mathcal{P}(\overline{1, n}) \mid \forall i \in I \forall j \in \overline{1, n} (j <_P i) \rightarrow (j \in I) \right\}$$

- Проверка $K \in \mathcal{P}(\overline{1, n})$ на «идеальность» перебором по определению: $\mathcal{O}(n^2)$.
- Просеивание $\mathcal{P}(\overline{1, n})$ займет $\mathcal{O}(n^2 \cdot 2^n)$ – долго.

Лучше взять другую процедуру, близкую к (Lawler, 1979), (Ченцов, 2004):

$$\mathcal{I}_{k+1} = \{ I \cup \{m\} \mid (I \in \mathcal{I}_k) \wedge (m \in \mathbf{E}[I]) \}.$$

Получим все $I \in \mathcal{I}$ за $\mathcal{O}(wn \cdot |\mathcal{I}|)$ (в литературе известно $\mathcal{O}(n^2 \cdot |\mathcal{I}|)$):

- Для каждого I за $\mathcal{O}(n)$ посчитаем $\mathbf{E}[I]$
- Можем повториться, но не более w раз, из каждого

$$I' \in \mathcal{I}_{|I|-1} \text{ такого что } \exists y \in \mathbf{I}[I] \mid I' = I \setminus \{y\}$$

$\mathbf{E}[K]$ за $\mathcal{O}(n)$

$$\uparrow x \triangleq \{m \in \overline{1, n} \mid m >_P x\}$$

$$\text{Min}[K] = K \setminus \bigcup_{x \in K} \uparrow x.$$

$\mathbf{I}[K]$ за $\mathcal{O}(n)$

$$\downarrow x \triangleq \{m \in \overline{1, n} \mid m <_P x\}$$

$$\text{Max}[K] = K \setminus \bigcup_{x \in K} \downarrow x.$$

Процедура ДП

Операция $\oplus$ должна быть *левоассоциативной* и *неубывающей* по *первому* аргументу (для прямой процедуры)

Цена состояния / функция Беллмана

$$v(K, x) \triangleq \begin{cases} c(0, x), & K = \emptyset; \\ \min_{m \in I[K]} \{v(K \setminus \{m\}, m) \oplus c(m, x)\}, & \text{otherwise.} \end{cases}$$

Состояния — идеалы с *выходом*

$$\begin{aligned} \mathcal{S} \triangleq & \left\{ (\emptyset, x) \in \{\emptyset\} \times \overline{1, n} \mid x \in \text{Min}\{\overline{1, n}\} \right\} \cup \\ & \cup \left\{ (I, x) \in \mathcal{I} \times \overline{1, n} \mid (I \in \mathcal{I}) \wedge (x \in \mathbf{E}[I]) \right\} \cup \left\{ (\overline{1, n}, \mathfrak{t}) \right\}; \end{aligned}$$

Слои пространства состояний: $\mathcal{S}_k \triangleq \{(K, x) \in \mathcal{S} \mid |K| = k\}$;

$$\mathcal{S} = \mathcal{S}_0 \sqcup \dots \sqcup \mathcal{S}_n$$

Псевдокод ДП

```

1: Инициализация:  $\mathcal{I}_0 = \{\emptyset\}$ ,  $S_0 = \{(\emptyset, x), x \in \mathbf{E}[\emptyset]\}$ ,
2:  $\forall (\emptyset, x) \in S_0 \ v(\emptyset, x) = c(\emptyset, 0, x)$ ,  $\mathcal{I}_1 = \mathbf{E}[\emptyset]$ .
3: for all  $k \in \overline{1, n-1}$  do
4:   for all  $I \in \mathcal{I}_k$  do
5:     Вычислить  $\mathbf{E}[I]$ 
6:     for all  $x \in \mathbf{E}[I]$  do
7:       Добавить  $I \cup \{x\}$  в  $\mathcal{I}_{k+1}$ 
8:       Добавить  $(I, x)$  в  $S_k$ 
9:       Вычислить  $v(I, x)$ 
10: Вычислить  $v(\overline{1, n}, \mathbb{t})$ 
11: Восстановить решение по  $v[S]$ .

```

Псевдокод **усеченного** ДП / Restricted DP

$H \in \mathbb{N}$ — параметр эвристики, количество *сохраняемых* состояний

- 1: **Инициализация:** $\tilde{\mathcal{I}}_0 = \{\emptyset\}$, $\tilde{\mathcal{S}}_0 = \{(\emptyset, x), x \in \mathbf{E}[\emptyset]\}$,
- 2: $\forall (\emptyset, x) \in \tilde{\mathcal{S}}_0 \ \tilde{v}(\emptyset, x) = c(\emptyset, 0, x)$, $\tilde{\mathcal{I}}_1 = \mathbf{E}[\emptyset]$.
- 3: **for all** $k \in \overline{1, n-1}$ **do**
- 4: **for all** $I \in \tilde{\mathcal{I}}_k$ **do** ▷ для всех *сохраненных* на шаге k
- 5: Вычислить $\mathbf{E}[I]$
- 6: **for all** $x \in \mathbf{E}[I]$ **do**
- 7: Вычислить $\tilde{v}(I, x)$ ▷ $\tilde{v}(I, x) \geq v(I, x)$, часть покрываемых могла *не сохраниться*
- 8: **if** $|\tilde{\mathcal{S}}_k| < H$ **then**
- 9: Добавить (I, x) в $\tilde{\mathcal{S}}_k$
- 10: **else**
- 11: Пусть $(J, y) = \operatorname{argmax}_{s \in \tilde{\mathcal{S}}_k} \tilde{v}(s)$
- 12: **if** $\tilde{v}(J, y) > \tilde{v}(I, x)$ **then**
- 13: Удалить (J, y) из $\tilde{\mathcal{S}}_k$
- 14: Добавить (I, x) в $\tilde{\mathcal{S}}_k$
- 15: **for all** $(I, x) \in \tilde{\mathcal{S}}_k$ **do**
- 16: Добавить $I \cup \{x\}$ в $\tilde{\mathcal{I}}_{k+1}$
- 17: Вычислить $\tilde{v}(\overline{1, n}, \mathfrak{t})$
- 18: **Восстановить решение по** $\tilde{v}[\tilde{\mathcal{S}}]$.

Средства

Язык C++11 — `std::unordered_mapa`, `std::chronob`

Параллелизм OpenMP 3.0 — конструкции `task`, для обработки хеш-таблицы

Среда Суперкомпьютер «Уран»

^aхеш-таблица

^bкроссплатформенный секундомер

Типы и структуры данных

множество: `std::uint32_t / std::bitset`

номер города: `std::uint16_t`

цена: `float / std::int32_t`

основная **структура данных:**

```
std::vector< std::unordered_map< std::bitset,  
std::unordered_map<std::uint16_t,t_cost> > >
```

доступ к $v(l, x)$: `foo[l][l][x]`

task-Параллелизация обработки хеш-таблицы

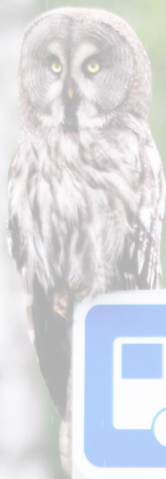
```

1: Инициализация:  $\mathcal{I}_0 = \{\emptyset\}$ ,  $\mathcal{S}_0 = \{(\emptyset, x), x \in \mathbf{E}[\emptyset]\}$ ,
2:  $\forall (\emptyset, x) \in \mathcal{S}_0 \ \forall (\emptyset, x) = \mathbf{c}(\emptyset, 0, x)$ ,  $\mathcal{I}_1 = \mathbf{E}[\emptyset]$ .
3: for all  $k \in \overline{1, n-1}$  do
4: #pragma omp parallel default (shared)
5: #pragma omp single nowait
6:   for all  $I \in \mathcal{I}_k$  do
7:     #pragma omp task untied firstprivate(I)
8:     Вычислить  $\mathbf{E}[I]$ 
9:     for all  $x \in \mathbf{E}[I]$  do
10:      #pragma omp critical (expand)
11:        Добавить  $I \cup \{x\}$  в  $\mathcal{I}_{k+1}$ 
12:        Добавить  $(I, x)$  в  $\mathcal{S}_k$ 
13:      Вычислить  $v(I, x)$ 
14:      #pragma omp critical (costwrite)
15:       $\text{foo}[|I|][I][x] := v(I, x)$ 
16: Вычислить  $v(\overline{1, n}, \mathbf{t})$ 
17: Восстановить решение по  $v[\mathcal{S}]$ .

```

- почти Линейное масштабирование $1 \rightarrow 8$ ядер в (SD)-BGTSP-PC
- Нет выигрыша для **не**обобщенной задачи в **целых** числах

Сложность



Пространственная — число состояний

$$0 \longrightarrow \overline{1, n} \longrightarrow t$$

$\underbrace{\hspace{10em}}_{\mathcal{C}[\cdot] \rightarrow \min}$

TSP

$$|S| = n2^{n-1} + 1$$

(Held, Karp, 1962)

TSP-PC

$$|S| \leq w|\mathcal{I}| \text{ (Sali, 2017)}$$

Ширина частичного порядка: $w \triangleq \max_{A \in AC} \{|A|\}$, где AC — множество *антицепей*

- подсчет ширины: через *максимальное паросочетание* во вспомогательном *двудольном графе* ($n + n$ вершин; ребра — элементы $<_p$)
- подсчет числа *порядковых идеалов* столь же труден как подсчет *антицепей*
 $\#P$ -труден^a (Provan, Ball, 1983); как NP -труден, только для *перечислительных задач*

^a «КР-труден» в (Гэри, Джонсон, 1982)

Временная сложность — *полином* от числа идеалов

$$0 \longrightarrow \underbrace{\overline{1, n}}_{\mathfrak{C}[\cdot] \rightarrow \min} \longrightarrow \mathfrak{t}$$

$$v(K, x) \triangleq \min_{m \in I[K]} \{v(K \setminus \{m\}, m) \oplus c(m, x)\} \quad (\text{BF})$$

TSP

$$\mathcal{O}(n^2 \cdot 2^{n-1})$$

(Held, Karp, 1962)

TSP-PC

$$\mathcal{O}(wn \cdot |\mathcal{I}|)$$

(Salii, 2017)

$\mathcal{O}(wn)$ нужно на генерацию^a каждого идеала

$\mathcal{O}(wn)$ нужно на расчет^b функции Беллмана состояний, соответствующих каждому $K \in \mathcal{I}$;

$\mathcal{O}(wn)$ уйдет на восстановление оптимального решения — в худшем случае

^a все теоретико-множественные операции выполняем за $\mathcal{O}(1)$; верно, когда множество укладывается в l -разрядное слово, иначе $\mathcal{O}(\lceil n/l \rceil)$

^b вычисляем $v(K \setminus \{m\}, m) \oplus c(m, x)$ за $\mathcal{O}(1)$; в числах с плавающей точкой константа **будет больше**

Временная сложность **усеченного** ДП

не более H , $H \in \mathbb{N}$, состояний на каждом слое

$$0 \longrightarrow \underbrace{1, n}_{\mathfrak{c}[\cdot] \rightarrow \min} \longrightarrow \mathfrak{t}$$

$$v(K, x) \triangleq \min_{m \in I[K]} \{v(K \setminus \{m\}, m) \oplus \mathfrak{c}(m, x)\} \quad (\text{BF})$$

(TD)-TSP

$$\mathcal{O}(n^2 \cdot H \cdot \log_2 H)$$

по (Malandraki, Dial, 1996)

TSP-PC

$$\mathcal{O}\left(n \cdot \min\left\{\frac{n+1}{2}; w\right\} \cdot H \cdot (nw + \log_2 H)\right).$$

(Salii, 2017)

Проблема

Нельзя просто так взять



И сосчитать^a все идеалы

^aэквивалент #P-трудной задачи АНТИЦЕПЬ (Provan, Ball, 1983)

Оценка числа идеалов

Оценки на основе ширины:

$$2^w + n - w \leq \mathcal{I} \leq \prod_{i=1}^k |C_i| \leq \left(\frac{n+w}{w}\right)^w$$

Удобнее рассматривать их *двоичный логарифм* — легче оценивать *расход памяти*.



[George Steiner.](#)

On the complexity of dynamic programming for sequencing problems with precedence constraints.

Annals of Operations Research, 26(1):103–123, 1990.



[George Steiner.](#)

On estimating the number of order ideals in partial orders, with some applications.

Journal of statistical planning and inference, 34(2):281–290, 1993.

TSPLIB–SOP: Сложностные характеристики I

NAME ^a	n	RED.	CL.	DENSITY	WIDTH	LOG_S-LB	LOG_S-UB
br17.10.sop	16	10	15	0,12	10	13,4	17,2
br17.12.sop	16	12	22	0,18	9	12,2	16,5
ESC07.sop	7	6	7	0,33	5	7,5	8,7
ESC11.sop	11	3	5	0,09	9	12,2	13,6
ESC12.sop	12	7	11	0,17	10	13,4	14,7
ESC25.sop	25	9	11	0,04	19	23,3	27,3
ESC47.sop	47	10	32	0,03	41	46,4	50,6
ESC63.sop	63	95	233	0,12	53	58,8	65,7
ESC78.sop	78	77	283	0,09	32	37,1	62,1
ft53.1.sop	52	12	12	0,01	42	47,4	54,3
ft53.2.sop	52	25	30	0,02	34	39,1	50,7
ft53.3.sop	52	48	217	0,16	24	28,6	44,5
ft53.4.sop	52	63	759	0,57	13	16,8	33,9
ft70.1.sop	69	17	17	0,01	55	60,8	70,3
ft70.2.sop	69	35	48	0,02	44	49,5	65,4
ft70.3.sop	69	68	215	0,09	35	40,2	60,2
ft70.4.sop	69	86	1 325	0,56	16	20,1	42,6
kro124p.1.sop	99	25	33	0,01	78	84,3	98,5
kro124p.2.sop	99	49	68	0,01	65	71,1	92,9
kro124p.3.sop	99	97	266	0,05	43	48,5	79,6
kro124p.4.sop	99	131	2 305	0,48	22	26,5	58,6

^aкрасные «ДП-разрешимы»; для полужирных неизвестно опт. решение

TSPLIB–SOP: Сложностные характеристики II

NAME ^a	n	RED.	CL.	DENSITY	WIDTH	LOG_S-LB	LOG_S-UB
p43.1.sop	42	9	11	0,01	36	41,2	45,4
p43.2.sop	42	20	34	0,04	26	30,8	40,8
p43.3.sop	42	37	96	0,11	21	25,4	37,7
p43.4.sop	42	50	496	0,58	13	16,8	30,8
prob.100.sop	98	41	41	0,01	57	62,9	88,1
prob42.sop	40	10	19	0,02	34	39,1	43,3
rbg048a.sop	48	192	447	0,40	32	37,1	47,4
rbg050c.sop	50	256	508	0,41	31	36	48
rbg109a.sop	109	622	5 329	0,91	12	15,7	43,6
rbg150a.sop	150	952	10 334	0,92	13	16,8	51,2
rbg174b.sop	174	1 113	13 955	0,93	22	26,5	73,9
rbg253a.sop	253	1 721	30 181	0,95	22	26,5	84,7
rbg323a.sop	323	2 412	48 202	0,93	47	52,6	145,5
rbg341a.sop	341	2 542	54 303	0,94	33	38,1	120,7
rbg358a.sop	358	3 239	56 536	0,88	55	60,8	165,8
rbg378a.sop	378	3 069	63 585	0,89	55	60,8	169,6
ry48p.1.sop	47	11	12	0,01	37	42,3	49
ry48p.2.sop	47	23	26	0,02	29	33,9	45,2
ry48p.3.sop	47	42	132	0,12	19	23,3	38,4
ry48p.4.sop	47	58	596	0,55	12	15,6	31,2

^aкрасные «ДП-разрешимы»; для полужирных неизвестно опт. решение

Экземпляры задачи, решаемые точным ДП

NAME	MEM(MB)	TSP-PC		TD-SOP	
		VALUE	RUN TIME	VALUE	RUN TIME
br17.10.sop	9,07	55	0,082	461	0,077
br17.12.sop	8,06	55	0,046	461	0,046
ESC07.sop	7,32	2 125	0,013	7 825	0,011
ESC11.sop	7,59	2 075	0,022	11 686	0,025
ESC12.sop	7,32	1 675	0,034	11 158	0,029
ESC25.sop	2 059.26	1 681	99	17 752	99
ft53.4.sop	73,88	14 425	2,035	383 057	2,196
ft70.4.sop	968,88	53 530	34	1 950 951	34
p43.3.sop	45 756.44	28 835	2 549	455 810	2 628
p43.4.sop	22,67	83 005	0,665	109 3510	0,46
rbg109a.sop	13,12	1 038	0,345	59 115	0,301
rbg150a.sop	18,54	1 750	0,715	114 753	0,512
rbg174b.sop	2 972.5	2 033	177	180 869	177
rbg253a.sop	3 089.98	2 950	192	370 847	192
ry48p.3.sop	128 432.31	1 984	8 331	461 275	8 331
ry48p.4.sop	34,94	31 446	0,696	736 631	0,896

Заключение.
Дальнейшие
исследования



Заключение

- 1 ДП **пригодно** для решения экземпляров (SD)-TSP-PC с «сильными» условиями предшествования, *сравнимо* по эффективности с CP/MDD.

- 2 Сила условий предшествования зависит от их *ширины* w и может измеряться через *количество состояний*. Все задачи TSPLIB с логарифмом верхней оценки числа состояний менее 40 разрешимы точным ДП.
- 3 ДП **полиномиально** по количеству идеалов. (SD)-TSP-PC **полиномиально** разрешима при $|\mathcal{I}| \in \text{Poly}(n)$.
- 4 В расчетах с *плавающей запятой* нет потребности в генерации $\mathcal{I}$ быстрее чем за $\mathcal{O}(wn)$ на идеал.

- 5 Эвристика «усеченное ДП» (restricted DP) достаточно эффективна для задач с «сильными» и «средними» условиями предшествования. Зачастую достигает оптимального решения существенно быстрее точного ДП и с сильно меньшим расходом памяти.
 - Всегда находит **допустимое** решение (важно когда много ограничений)
 - Полиномиальна по H и n
 - Решение стремится к точному при приближении H к $\arg\max_{i \in \overline{0, n}} \{|\mathcal{S}_i|\}$

I. Вычисления: дальше и больше

- Схема ветвей и границ в ДП по (Morin, Marsten, 1976). Уже применялась к TSP-PC в (Bianco et al., 1992).
- Параллелизация ДП (Ченцов, 2012), (Салий, 2015).
- Разработка (и решение) экземпляров GTSP-PC. Вероятно, (а) оснастить известные GTSP-экземпляры случайными условиями предшествования, (б) кластеризовать задачи из TSPLIB с сохранением предшествований
- Гарантии качества для *усеченного* ДП.

II. Сложность задач с частичным порядком

- Улучшить оценку пространственной сложности с учетом (Habib, Medina, Nourine, Steiner, 2001)
- Исследовать теоретическую эффективность схемы параллелизации с «независимыми вычислениями» (Ченцов, 2012)
- Получить простые в вычислении оценки количества идеалов фиксированной мощности $\mathcal{I}_i = \{I : |I| = k\}$.
- Улучшить оценки сверху для $|\mathcal{I}|$; может быть, найти-таки *оптимальное* разложение в цепи

III. ДП и другие задачи с условиями предшествования

- Есть готовая теория для задачи об ориентировании (OP-PC), задачи коммивояжера с призами (PCTSP-PC) с условиями предшествования
- Можно рассмотреть задачу о рюкзаке с условиями предшествования (Kolliopoulos, Steiner, 2007)

Публикации по теме диссертации I



Ченцов Александр Георгиевич, Салий Ярослав Витальевич. *Об одной задаче на узкие места* // Сборник трудов Всероссийской **конференции** СМО-2011 (Челябинск, 28 ноября – 3 декабря 2011 г.)
2011. С. 85–93.



Салий Ярослав Витальевич. *Об ультраметрической задаче коммивояжера на узкие места* // Современные проблемы математики. Тезисы Международной (43-й Всероссийской) молодёжной школы-конференции / ИММ УрО РАН.
2012.
С. 287–289.



Салий Ярослав Витальевич, Ченцов Александр Георгиевич. *Об одной маршрутной задаче на узкие места с внутренними работами* // **Вестник Тамбовского университета. Серия: Естественные и технические науки**.
2012. Т. 17, № 3.



Салий Ярослав В, Ченцов Александр Г. *О маршрутной задаче на узкие места с внутренними работами и условиями предшествования* // Международная конференция «Дискретная оптимизация и исследование операций»: Материалы **конференции** (Новосибирск, 24 – 28 июня 2013 г.) / ИМ СО РАН, НГУ.
2013. С. 134.

Публикации по теме диссертации II



Салий Ярослав Витальевич. *Влияние условий предшествования на вычислительную сложность решения маршрутных задач методом динамического программирования* // **Вестник Удмуртского университета. Математика. Механика. Компьютерные науки.** 2014. № 1. С. 76–86.



Ченцов Александр Г, Салий Ярослав В. *Неаддитивная задача маршрутизации с условиями предшествования* // Алгоритмический анализ неустойчивых задач: тез. докл. Всерос. **конф.** с междунар. участием, посвящ. памяти В. К. Иванова, Челябинск, 10 – 14 нояб. 2014 г. / Южно-Уральский государственный Университет; ИММ УрО РАН; УрФУ. 2014. С. 166–167.



Chentsov A. G., Salii Y. V. *A model of “nonadditive” routing problem where the costs depend on the set of pending tasks* // **Vestnik YuUrGU. Ser. Mat. Model. Progr.** 2015. Vol. 8, no. 1. PP. 24–45.

Публикации по теме диссертации III



Салий Ярослав Витальевич. *О прямом и попятном динамическом программировании в маршрутных задачах с условиями предшествования и алгоритмах генерации допустимых подзадач* // Тезисы докладов XV Всероссийской конференции «Математическое программирование и приложения» / ИММ УрО РАН, УрФУ.

Информационный бюллетень Ассоциации математического программирования № 13.

2015. С. 168–169.



Ченцов Александр Георгиевич, Салий Ярослав Витальевич. *Задача маршрутизации «на узкие места» с ограничениями и усложненными функциями стоимости* // Материалы 8-й Всерос. мультиконф. (МКПУ-2015): в 3-х т. Ростов-на-Дону: Изд. ЮФУ, 2015. Т. 1. С. 214–216



Salii Y. V. *Restricted Dynamic Programming Heuristic for Precedence Constrained Bottleneck Generalized TSP* // Proceedings of the 1st Ural Workshop on Parallel, Distributed, and Cloud Computing for Young Scientists / Ed. by A. Sozykin, E. Akimova, D. Ustalov.

Vol. 1513. **CEUR Workshop Proceedings**, 2015. P. 85–108.

<http://ceur-ws.org/Vol-1513/#paper-10>.

Публикации по теме диссертации IV



Salii Y. V., Ivanko E. E. *Dynamic programming and greedy heuristic in Camera Trap Traveling Salesman Problem* // Современные проблемы математики и ее приложений.

Vol. 1662. **CEUR Workshop Proceedings**, 2016. P. 215–219.

<http://ceur-ws.org/Vol-1662/>.



Salii Y. V. *Order-theoretic characteristics and dynamic programming for Precedence Constrained Traveling Salesman Problem* // Proceedings of the Fourth Russian Finnish Symposium on Discrete Mathematics / Ed. by J. Karhumäki, Y. Matiyasevich, A. Saarela.

Vol. 26 of *TUCS Lecture Notes*.

Turku Centre for Computer Science, 2017. P. 152–164.

<http://urn.fi/URN:ISBN:978-952-12-3547-4>.



Salii Y. V. *Revisiting Dynamic Programming for Precedence-Constrained Traveling Salesman Problem and Its Time-Dependent Generalization*.

Revised 16.10.2017, under review in European Journal of Operational Research; preprint available at Researchgate, [https:](https://www.researchgate.net/profile/Yaroslav_Salii/publications)

[//www.researchgate.net/profile/Yaroslav_Salii/publications](https://www.researchgate.net/profile/Yaroslav_Salii/publications)

Апробирование — выступления I



Ченцов Александр Георгиевич, Салий Ярослав Витальевич. *Об одной задаче на узкие места* // Всероссийская **конференция** СМО-2011 (Челябинск, 28 ноября – 3 декабря 2011 г.)



Салий Ярослав Витальевич. *Об ультраметрической задаче коммивояжера на узкие места* // Современные проблемы математики. Международная (43-я Всероссийская) молодёжная школа-**конференция** (Екатеринбург, 2012)



Салий Ярослав В, Ченцов Александр Г. *О маршрутной задаче на узкие места с внутренними работами и условиями предшествования* // Международная конференция «Дискретная оптимизация и исследование операций»: Материалы **конференции** (Новосибирск, 24 – 28 июня 2013 г.)



Ченцов Александр Г, Салий Ярослав В. *Неаддитивная задача маршрутизации с условиями предшествования* // Алгоритмический анализ неустойчивых задач: тез. докл. Всерос. **конф.** с междунар. участием, посвящ. памяти В. К. Иванова (Челябинск, 2014)



Салий Ярослав Витальевич. *О прямом и попятном динамическом программировании в маршрутных задачах с условиями предшествования и алгоритмах генерации допустимых подзадач* // XV Всероссийская **конференция** «Математическое программирование и приложения» (Екатеринбург, 2015)

Апробирование — выступления II



Ченцов Александр Георгиевич, Салий Ярослав Витальевич. *Задача маршрутизации «на узкие места» с ограничениями и усложненными функциями стоимости* // 8-й Всерос. **мультиконф.** (Дивноморское, 2015)



Salii Y. V. *Restricted Dynamic Programming Heuristic for Precedence Constrained Bottleneck Generalized TSP* // 1st Ural **Workshop** on Parallel, Distributed, and Cloud Computing for Young Scientists (Yekaterinburg, 2015)



Salii Y. V., Ivanko E. E. *Dynamic programming and greedy heuristic in Camera Trap Traveling Salesman Problem* // Международная (46-я Всероссийская) конференция **Современные проблемы математики и ее приложений** (Екатеринбург, 2016).



Салий Я. В. *Испытание эвристики усеченного динамического программирования на известных экземплярах задачи коммивояжера с условиями предшествования* // Международная (47-я Всероссийская) конференция **Современные проблемы математики и ее приложений** (Екатеринбург, 2017).



Salii Y. V. *Order-theoretic characteristics and dynamic programming for Precedence Constrained Traveling Salesman Problem* // Fourth Russian Finnish **Symposium** on Discrete Mathematics (Turku, 2017)

Свидетельства о регистрации



Я.В. Салий, Программа вычисления оптимального решения обобщенной задачи курьера на узкие места с условиями предшествования и зависимостью от списка невыполненных заданий методом динамического программирования // Федеральная служба по интеллектуальной собственности, (12) государственная регистрация программы для ЭВМ. — № 2015661435. — дата регистрации 27.10.2015



Программа расчета сложности задачи о маршрутизации инструмента в машинах листовой резки с целью сокращения холостого хода // Федеральная служба по интеллектуальной собственности, (12) государственная регистрация программы для ЭВМ. — № 2015661434. — дата регистрации 27.10.2015