

Spis treści

1	Wstęp	3
2	Przegląd literatury	5
2.1	Przegląd - notatki	5
2.2	O przestrzeni rozwiązań	8
2.3	Sieć optimów lokalnych	8
3	Badania eksperymentalne	11
3.1	Opis eksperymentów	11
3.2	Zaimplementowane algorytmy	11
3.3	Wyniki	11
4	Opis implementacji	13
5	Podsumowanie	15
	Dodatki	17
A	Jakiś dodatek	19
	Literatura	19
	Spis rysunków	21
	Spis tablic	23

Rozdział 1

Wstęp

To jest test bibtexa [4] praca [5] na pewno się przyda. blablabla wstęp [1]

Rozdział 2

Przegląd literatury

2.1 Przegląd - notatki

Mapping the global structure of TSP fitness landscapes(2018)[3]

W pracy przedstawiony został algorytm próbkujący przeszukujący przestrzeń rozwiązań problemu komiwojażera(TSP). Algorytm konstruuje sieć optimów lokalnych(LON) - odmianę z tzw. escape edges. Do próbkowania wykorzystywana jest implementacja Concorde algorytmu Chained-Lin-Kerningham. Wykonane zostało 1000 niezależnych uruchomień (przebiegów?) algorytmu. Połowa była inicjowana metodą Quicka-Borůvki, a druga połowa losowym rozwiązaniem. Algorytm opiera się na generowaniu nowych rozwiązań poprzez zastosowanie perturbacji typu double-bridge, a następnie uruchomieniu algorytmu Lin-Kerningham, by znaleźć lokalne minimum. Operacja jest powtarzana 10000 razy i budowana jest w ten sposób sieć optimów lokalnych.

Jednym z zadań, które postawili sobie autorzy pracy, jest identyfikacja "struktur lejowych" w przestrzeni rozwiązań. Lej jest sekwencją lokalnych optimów, których wartość dopasowania się nie pogarsza, i która zmierza do pewnego optimum - globalnego (primary funnel) lub lokalnego (secondary funnels). To optimum nazywane jest (spodem leja?) funnel floor/bottom. Poszczególne sieci optimów lokalnych wygenerowane przez kolejne uruchomienia algorytmu próbkującego zawyżały liczbę lejów w przestrzeni rozwiązań. Badacze scalili więc znalezione optima lokalne z wszystkich przebiegów algorytmu. Znalezione w ten sposób rozwiązania nazwali atraktorami i utworzyli graf sieci atraktorów, w której są one wierzchołkami. Następnie funnel bottoms zostały zidentyfikowane jako wierzchołki, z których nie ma krawędzi wychodzących. Następnie przynależność optimów lokalnych do poszczególnych lejów została sprawdzona poprzez sprawdzenie, czy istnieje ścieżka w grafie pomiędzy danym optimum a spodem leja.

Do badań wykorzystano 20 instancji TSP o rozmiarach od 500 do 1500 miast. Wykorzystano instance o miastach rozłożonych równomiernie i w klastrach. Część została wygenerowana, a część pochodzi z TSPLIB.

Zbadano właściwości powstałych sieci optimów lokalnych, m.in Liczbę unikalnych optimów lokalnych i globalnych, liczba wykonań funkcji celu, stosunek liczby wierzchołków i wywołań funkcji celu, jaka część wierzchołków jest połączona ścieżką z optimum globalnym, średnia siła(strength) wierzchołków. Zbadano również własności lejów, takie jak rozmiar największego leja czy liczba atraktorów.

Badania wykazały, że przestrzeń rozwiązań badanych instancji TSP zawiera wiele lejów a nie, jak wcześniej sądzono, jeden lej kończący się w globalnym minimum. W przestrzeni rozwiązań instancji o regularnej strukturze wykryto istnienie rozległych "płaskowy-

zów". Zauważono różnicę między instancjami wygenerowanymi losowo a rzeczywistymi. Te pierwsze miały zwykle jedno globalne optimum i małą neutralność, podczas gdy instancje rzeczywiste zwykle miały więcej globalnych optimów, czasem formujących płaskowyż globalnych optimów. Zauważono zależność między rozmiarem instancji a liczbą lejów - ich liczba zwiększa się wraz ze wzrostem rozmiaru instancji. Najwięcej lejów pojawiło się w instancjach z równomiernym losowym rozmieszczeniem miast. Zbadano korelację między różnymi właściwościami przestrzeni rozwiązań. Wykazano, że w instancjach, w których miasta rozmieszczone są w klastrach, występuje większa korelacja właściwości, niż w instancjach o rozłożeniu równomiernym.

Local Optima Networks in Solving Algorithm Selection Problem for TSP(2018)[1]

W pracy badano możliwość wykorzystania analizy sieci optimów lokalnych w problemie wyboru algorytmu heurystycznego dla danej instancji problemu komiwojażera (Algorithm selection problem).

Algorytm próbkowania podzielony jest na dwa etapy - poszukiwanie wierzchołków i poszukiwanie krawędzi. Wierzchołki wyszukiwane są poprzez generowanie losowych rozwiązań i uruchamianie na nich algorytmu 2-opt. Próbkowanie krawędzi polega na poddaniu każdego ze znalezionych wcześniej lokalnych optimów perturbacji, optymalizacji 2-opt typu first-improvement i dodaniu odpowiedniej krawędzi lub zwiększenia jej wagi jeśli już istnieje. Wśród badanych parametrów wymienić można m.in stosunek liczby krawędzi do liczby wierzchołków, średnia odległość optimum lokalnego od optimum o najniższej wartości funkcji celu, stosunek wierzchołków połączonych z tym optimum do pozostałych, oraz różnorodność (assortativity) i współczynnik klasteryzacji grafu. Do eksperymentów wykorzystano instancje z biblioteki TSPLIB, oraz instancje generowane losowo o równomiernym rozkładzie miast. Obliczono korelację różnych miar, aby wyeliminować redundantne dane.

Na każdej instancji problemu uruchomiono algorytmy heurystyczne i porównano parami jakość ich rozwiązań. Obliczono również średnią względną jakość rozwiązań dla każdego algorytmu. Porównane zostały algorytmy Greedy Descent, Guided Local Search, Simulated Annealing, Tabu Search, Objective Tabu Search, zaimplementowane w pakiecie Google Optimization Tools. Sprawdzono również obecny w pakiecie tryb automatyczny, który sam dobiera algorytm dla danej instancji.

Na podstawie zebranych danych nauczono 3-klasowe klasyfikatory dla każdej pary algorytmów. (algorytm A daje lepszy wynik od algorytmu B, algorytm B daje lepszy wynik od algorytmu A, algorytmy dają ten sam wynik). Utworzone klasyfikatory porównano z klasyfikatorem ZeroR. Przygotowane klasyfikatory dokonały poprawnej predykcji częściej niż ZeroR w przypadku instancji ze zbioru TSPLIB. W przypadku instancji generowanych losowo, tylko w jednym przypadku klasyfikator okazał się lepszy od ZeroR.

Eksperymenty wykazały, że analiza sieci optimów lokalnych mogą zostać z powodzeniem wykorzystana do selekcji algorytmu heurystycznego, jednak długi czas próbkowania niweluje korzyści uzyskane w ten sposób.

Clarifying the Difference in Local Optima Network Sampling Algorithms(2019)

W pracy wykonane zostało statystyczne porównanie dwóch algorytmów próbkowania sieci lokalnych optimów dla kwadratowego problemu przydziału (QAP). Do badań wykorzystano instancje problemu z biblioteki QAPLIB. Wybrano 30 instancji o różnych klasach (rozłożone równomiernie, na siatce, instancje rzeczywiste, instancje losowe symulujące

rzeczywiste). Porównano algorytmy Snowball oraz Iterated Local Search. Spróbkowane dane wykorzystano do stworzenia modeli regresji do przewidywania jakości rozwiązań dwóch algorytmów heurystycznych: Robust Taboo Search Taillarda oraz Improved ILS Stützla.

Wykonano próbkowanie przestrzeni rozwiązań i sprawdzono, czy wartości poszczególnych miar pochodzące od dwóch algorytmów są ze sobą skorelowane. Zbadanymi miarami były m.in: liczba optimów, liczba krawędzi, średnia liczba krawędzi wychodzących z wierzchołka grafu (outdegree), średnica grafu, średnie dopasowanie lokalnych optimów (meanfitness), sinkfitness (dopasowanie spływów? TODO: translate). Zauważono korelację metryk meanfitness i sinkfitness na poziomie 0.99. Dla wszystkich innych metryk korelacja było mniejsza od 0.5, dla średnicy grafu była ona negatywna. Na podstawie zebranych danych nauczono modele typu liniowego oraz RandomForest. Zbadano wartości metryki R^2 dla różnych parametrów algorytmów próbkowania, algorytmów heurystycznych i modeli regresji. Obliczono również współczynniki korelacji pomiędzy różnymi właściwościami sieci lokalnych optimów a performance gap metric (TODO: translate) algorytmów optymalizujących.

Badania wykazały, że sieci wygenerowane przez dwa różne algorytmy próbkujące mają pewne wspólne cechy. Snowball okazał się bardziej przewidywalny i łatwiejszy w dobieraniu parametrów. ITS natomiast lepiej znajdował "struktury-huby" (TODO: popraw?) w przestrzeni rozwiązań. Wartości metryk LON uzyskane na podstawie algorytmu ITS były bardziej skorelowane z heuristic performance metric (TODO: translate). Modele regresji utworzone na podstawie danych uzyskanych z ITS dokonywały lepszej predykcji jakości rozwiązań algorytmów heurystycznych, niż modele utworzone na podstawie algorytmu Snowball. Zauważono, że wartości funkcji celu w poszczególnych lokalnych optimach sieci LON były lepszymi predyktorami, niż informacje o krawędziach tej sieci.

Inferring Future Landscapes: Sampling the Local Optima Level(2020)[6]

W pracy zostały porównane algorytmy próbkowania sieci optimów lokalnych(LON) dla kwadratowego problemu przydziału (QAP). Porównane zostały algorytmy Markov-Chain oraz Snowball. Dla małych instancji problemu wykonano kompletne przeszukanie przestrzeni rozwiązań, Dla większych - przeszukiwanie z limitowanym budżetem obliczeniowym.

Jako instancje testowe wykorzystano przykłady ze zbioru QAPLIB. Instancje należały do pięciu kategorii: Instancje z wartościami odległości i przepływu losowymi z rozkładu równomiernego, instancje z wartościami odległości ułożonymi "na siatce" i losowymi wartościami przepływu, rzeczywiste instancje problemu, instancje generowane symulujące rzeczywiste, oraz instancje nie pasujące do żadnej z poprzednich kategorii.

Badane właściwości sieci LON różniły się w zależności od algorytmu próbkowania. Dla Algorytmu Markov-chain były to: liczba wierzchołków i krawędzi grafu, średnie dopasowanie(mean fitness), średnia liczba krawędzi wychodzących z wierzchołka (mean out-degree), oraz najdłuższa ścieżka między wierzchołkami. Dodatkowo zbadano tzw. Funnel metrics - liczba krawędzi do globalnego optimum (incoming global) oraz ???(mean funnel-floor fitness)

Dla algorytmu snowball metryki funnel, oraz te oparte o gęstość i wzory połączeń krawędzi nie dają użytecznych informacji, zamiast tych metryk wybrano: średnią wagę pętli w grafie, średnią różnicę wag krawędzi wychodzących (mean weight disparity), korelacja dopasowania między sąsiadami (fitness-fitness correlation), średnia długość "wspinaczki" do optimum lokalnego, maksymalna długość "wspinaczki" do lokalnego optimum oraz

maksymalna liczba ścieżek prowadzących do lokalnego optimum.

Na podstawie danych zbudowano modele regresji, dokonujące predykcji jakości rozwiązań uzyskanych przy wykorzystaniu heurystyk Improved Iterated Local Search i Robust Taboo Search. Sprawdzono modele liniowe i random-forest. Random-forest okazały się lepsze.

Badania nie wykazały wyższości jednego algorytmu próbkowania nad drugim. Autorzy zwracają uwagę na fakt, że niektóre z cech sieci optimów lokalnych są lepiej odwzorowywane przez algorytm Markov-chain, a inne lepiej przez Snowball. Najlepsze wśród zbudowanych modeli regresji okazały się zaś te, które korzystały z kombinacji danych pozyskanych z obu algorytmów.

2.2 O przestrzeni rozwiązań

Fitness landscape (Przestrzeń rozwiązań? Krajobraz dopasowania?) jest pojęciem wywodzącym się z biologii ewolucyjnej. W kontekście biologicznym jest to model opisujący relację między genotypem i fenotypem organizmów, a ich przystosowaniem (ang. fitness), które jest miarą opisującą sukces reprodukcyjny[2].

W kontekście optymalizacji Fitness landscape to trójka (S, V, f) , gdzie:

- S jest zbiorem wszystkich możliwych rozwiązań - przestrzenią przeszukiwania,
- V jest funkcją przypisującą każdemu rozwiązaniu $s \in S$ zbiór sąsiadów $V(s)$,
- f jest funkcją $f : S \rightarrow \mathbb{R}$ przypisującą danemu rozwiązaniu wartość przystosowania - zwykle jest to wartość funkcji celu dla danego rozwiązania.

2.3 Sieć optimów lokalnych

Sieć optimów lokalnych (ang. Local Optima Network, LON) jest konstruktem zaprezentowanym po raz pierwszy w artykule[7], i rozwiniętym w pracy[4].

Jest to graf $G = (N, E)$ przedstawiający występujące w przestrzeni rozwiązań optima lokalne (zbiór wierzchołków N) i relacje między nimi (zbiór krawędzi E).

Wierzchołki

Wierzchołki w sieci optimów lokalnych reprezentują optima lokalne w przestrzeni rozwiązań. Do optimów zaliczamy minima i maksima; w problemach optymalizacyjnych zazwyczaj poszukujemy tych pierwszych. Minimum lokalne to takie rozwiązanie s , w którego sąsiedztwie $V(s)$ nie znajduje się żadne rozwiązanie x , dla którego $f(x) < f(s)$. Każde rozwiązanie w przestrzeni rozwiązań można przyporządkować do pewnego lokalnego minimum. Aby znaleźć lokalne minimum $n \in N$, do którego "prowadzi" dane rozwiązanie $s \in S$, wykonuje się lokalną optymalizację z tym rozwiązaniem przyjętym jako punkt startowy. W dalszej części pracy takie przyporządkowanie będzie oznaczane jako $h(s) \rightarrow n \in N$. Wierzchołkom w sieci LON można przypisać wagę równą wartości funkcji celu w danym optimum lokalnym.

Krawędzie

Powstały dwa modele krawędzi: basin-transition i escape edges (TODO: translate?)

Basin-transition

Basen przyciągania optimum lokalnego n jest zdefiniowany jako zbiór:

$$b_i = \{s \in S \mid h(s) = n\}$$

Rozmiarem basenu jest liczność tego zbioru oznaczana jako $\#b_i$. Dla każdej pary rozwiązań w przestrzeni można obliczyć prawdopodobieństwo przejścia z jednego rozwiązania do drugiego $p(s \rightarrow s')$. Dla rozwiązań reprezentowanych permutacją o długości M , prawdopodobieństwo takie wynosi:

$$p(s \rightarrow s') = \frac{1}{M(M-1)/2}, \quad \text{jeżeli } s' \in V(s),$$

$$p(s \rightarrow s') = 0, \quad \text{jeżeli } s' \notin V(s),$$

Mając informacje o prawdopodobieństwach przejścia między poszczególnymi rozwiązaniami można obliczyć prawdopodobieństwo przejścia od rozwiązania s do dowolnego rozwiązania należącego do basenu b_j :

$$p(s \rightarrow b_j) = \sum_{s' \in b_j} p(s \rightarrow s')$$

Całkowite prawdopodobieństwo przejścia z basenu jednego optimum lokalnego do drugiego wynosi więc:

$$p(b_i \rightarrow b_j) = \frac{1}{\#b_i} \cdot \sum_{s \in b_i} p(s \rightarrow b_j)$$

To całkowite prawdopodobieństwo stanowi wagę krawędzi w grafie.

Escape Edges

Escape Edges zdefiniowane są przy pomocy funkcji dystansu d zwracającej najmniejszą odległość między dwoma rozwiązaniami i liczby całkowitej N . Krawędź e_{ij} między lokalnymi optimami n_i i n_j istnieje, jeśli istnieje rozwiązanie s takie, że:

$$d(s, n_i) \leq D \wedge h(s) = n_j$$

Wagą takiej krawędzi jest ilość rozwiązań spełniających powyższy warunek.

TODO: napisać o różnicach sieci tworzonych przez basin-transition i escape edges

Rozdział 3

Badania eksperymentalne

3.1 Opis eksperymentów

3.2 Zaimplementowane algorytmy

3.3 Wyniki

Rozdział 4

Opis implementacji

Rozdział 5

Podsumowanie

Dodatki

Dodatek A

Jakiś dodatek

Literatura

- [1] W. BOZEJKO, A. GNATOWSKI, T. NIZYNSKI, M. AFFENZELLER, AND A. BEHAM, *Local optima networks in solving algorithm selection problem for TSP*, in Contemporary Complex Systems and Their Dependability - Proceedings of the 13th International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, July 2-6, 2018, Brunów, Poland, W. Zamojski, J. Mazurkiewicz, J. Sugier, T. Walkowiak, and J. Kacprzyk, eds., vol. 761 of Advances in Intelligent Systems and Computing, Springer, 2018, pp. 83–93.
- [2] I. FRAGATA, A. BLANCKAERT, M. A. DIAS LOURO, D. A. LIBERLES, AND C. BANK, *Evolution in the light of fitness landscape theory*, Trends in Ecology & Evolution, 34 (2019), pp. 69–82.
- [3] G. OCHOA AND N. VEERAPEN, *Mapping the global structure of TSP fitness landscapes*, J. Heuristics, 24 (2018), pp. 265–294.
- [4] G. OCHOA, S. VÉREL, F. DAOLIO, AND M. TOMASSINI, *Local optima networks: A new model of combinatorial fitness landscapes*, CoRR, abs/1402.2959 (2014).
- [5] S. L. THOMSON, G. OCHOA, AND S. VÉREL, *Clarifying the difference in local optima network sampling algorithms*, in Evolutionary Computation in Combinatorial Optimization - 19th European Conference, EvoCOP 2019, Held as Part of EvoStar 2019, Leipzig, Germany, April 24-26, 2019, Proceedings, A. Liefooghe and L. Paquete, eds., vol. 11452 of Lecture Notes in Computer Science, Springer, 2019, pp. 163–178.
- [6] S. L. THOMSON, G. OCHOA, S. VÉREL, AND N. VEERAPEN, *Inferring future landscapes: Sampling the local optima level*, Evol. Comput., 28 (2020), pp. 621–641.
- [7] M. TOMASSINI, S. VÉREL, AND G. OCHOA, *Complex-network analysis of combinatorial spaces: The nk landscape case*, Phys. Rev. E, 78 (2008), p. 066114.

Spis rysunków

Spis tabel