Problem komiwojażera dla symulowanego wyżarzania

referat

Kamil Król

244949

Spis treści

1. Opis problemu komiwojażera

1.1. Opis

Problem komiwojażera jest klasycznym przykładem problemu optymalizacyjnego. Należy on do problemów NP-trudnych, a oznacza to między innymi, że nie jest dotąd znany algorytm rozwiązujący ten problem w czasie wielomianowym. Zatem rozwiązanie tego problemu poprzez sprawdzenie wszystkich możliwości jest bardzo *kosztowne*. Mówiąc kolokwialnie – jest to problem bardzo trudny. Stąd właśnie wynika fakt, że to rozwiązania tego problemu często są używane mniej *kosztowne* heurystyki.
Problem polega na znalezieniu cyklu Hamiltona w pewnym ważonym grafie o jak najmniejszej sumie wag krawędzi. Problem często jest przedstawiany jako podróżowanie między miastami.
Formalnie:
Dane: lista miast (wierzchołków), odległości (wagi krawędzi) między każdymi dwoma miastami (wierzchołkami)
Szukane: najkrótsza ścieżka odwiedzająca każde miasto dokładnie raz powracająca do miasta startowego (cykl Hamiltona o minimalnej sumie wag).
Konieczne jest założenie, że rozważany graf jest spójny – z każdego miasta da się dojechać do dowolnego innego. Jeśli graf jest niespójny to cykl Hamiltona nie istnieje.

1.2. Matematyczne sformułowanie problemu

Cykl Hamiltona (cykl między miastami spełniający warunki opisane wyżej) oznaczany będzie przez c, jest to ciąg wierzchołków (miast) oddający kolejność w jakiej wierzchołki (miasta) są odwiedzane. Miasta

oznaczane są liczbami ze zbioru $\{1, 2, \ldots, n\}$, gdzie n to łączna liczba miast.

$$c = (c_1, c_2, c_3, \ldots, c_n) \text{ i jeśli } i \neq j \text{ to } c_i \neq c_j$$

$$c_i \in \{1, 2, \ldots, n\}$$

Zatem rozwiązanie problemu komiwojażera wiąże się z problemem minimalizacji funkcji:

$$F(c) = \sum_{i=1}^{n-1} (d_{c_i, c_j}) + d_{c_1, c_n}$$

gdzie d_{c_i, c_j} jest odległością między miastami i oraz j.

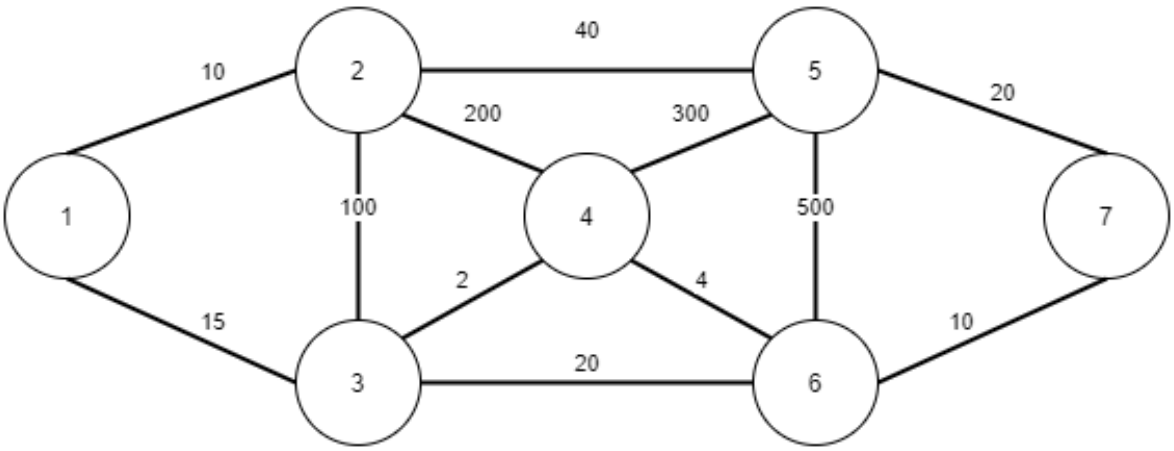
1.3. Przykład

Jest to mały przykład problemu komiwojażera. Na grafie poniżej (Rysunek 1.) można zauważyć kilka przykładowych cykli Hamiltona. Przez $c(x)$ oznaczmy sumę wag na cyklu x. Widać, że cykl τ_1 (Rysunek 2.) jest cyklem o najmniejszej sumie wag w tym grafie. Zatem można stwierdzić, że jest to optymalne rozwiązanie.

Przykładowe cykle Hamiltona w grafie na rysunku:

$$\tau_1 = 1 \to 2 \to 5 \to 7 \to 6 \to 4 \to 3 \to 1; \; c(\tau_1) = 101$$

$$\tau_2 = 1 \to 2 \to 4 \to 5 \to 7 \to 6 \to 3 \to 1; \; c(\tau_2) = 575$$



Rysunek 1: Przykładowy ważony graf nieskierowany

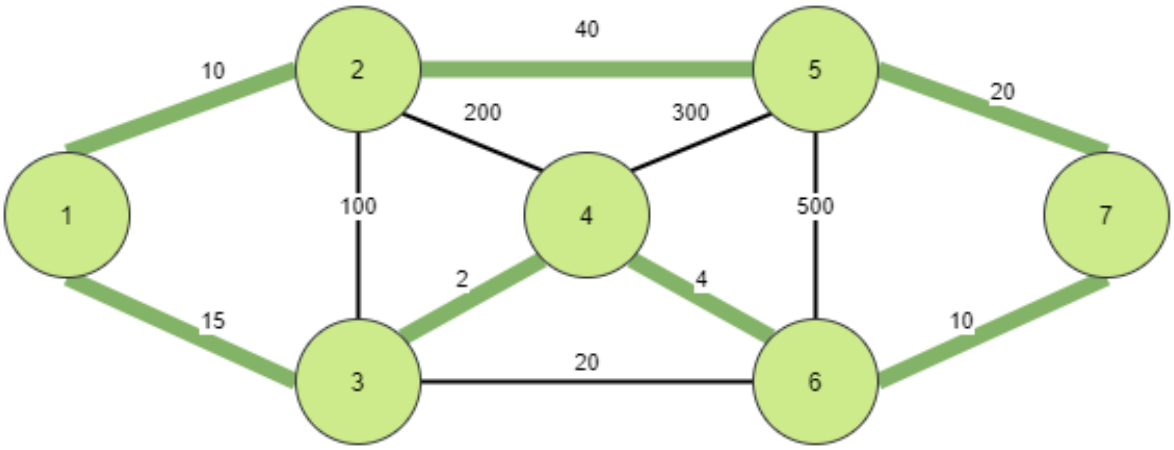
1.4. Możliwe modyfikacje

Problem ten może być modyfikowany na wiele różnych sposobów. Np. Skierowanie grafu (dopuszczenie dróg jednokierunkowych) nie zmienia istoty problemu ani jego klasy złożoności – problem nadal jest NP-trudny (o ile graf skierowany jest nadal spójny w sensie spójności w grafie skierowanym). Dodanie wymagania aby graf był grafem pełnym nic nie zmienia, ponieważ TSP w formie przedstawionej wyżej (graf niekoniecznie pełny) można sprowadzić do grafu pełnego poprzez dodanie krawędzi o nieskończonej wadze – nie zmieni to optymalnego rozwiązania.
Co stanie się jeżeli zrezygnujemy z konieczności powrotu do miasta startowego? Problem ten nadal jest NP-trudny. Aczkolwiek nie jest to już problem komiwojażera.

2. Symulowane wyżarzanie

2.1. Przypomnienie

Symulowane wyżarzanie jest metaheurystyką aproksymującą globalne minimum/maksimum pewnej danej funkcji. Jej idea oraz nazwa pochodzi od metody wyżarzania w metalurgii, w której to rozgrzewa się dany



Rysunek 2: Graf z rysunku 1. z zaznaczonym cyklem τ_1. Na zielono oznaczono krawędzie oraz wierzchołki przez które przechodzi cykl.

metal oraz stopniowo się go ochładza kontrolując stale temperaturę. Zbyt szybkie jego ochłodzenie nie da pożądanego efektu.
Dane: funkcja f do zoptymalizowania, rozwiązanie początkowe x_0, definicja sąsiedztwa $\mathcal{N}$, temperatura początkowa T_0, sposób ochładzania δ (np. liniowy), współczynni ochładzania β (jeśli potrzebny), funkcja P obliczająca prawdopodobieństwo zaakceptowania (z ang. acceptance probability) danego rozwiązania x w zależności od aktualnej temperatury T, energii aktualnego rozwiązania oraz energii danego x. Przez energię jakiegoś rozwiązania rozumiemy tutaj wartość funkcji w pewnym punkcie.
Metoda:

1. Za aktualne rozwiązanie x_c podstaw x_0 – rozwiązanie początkowe: $x_c \leftarrow x_0$,
2. Za aktualną temperaturę T podstaw T_0: $T \leftarrow T_0$,
3. Dopóki $T > 0$ wykonuj kroki:
   i) Weź nowego sąsiada aktualnego rozwiązania.
   Tzn. wybierz losowy element ze zbioru sąsiadów x_c: $x_n \leftarrow randomFrom(\mathcal{N}(x_c))$
   Wybrany x_n jest teraz kandydatem na nowe aktualne rozwiązanie,
   ii) Oblicz $p \leftarrow P(x_n, x_c, T)$ – prawdopodobieństwo zaakceptowania kandydata x_n,
   iii) Jeśli $p \geqslant uniform(0, 1)$ to podstaw: $x_c \leftarrow x_n$. Jeśli nie, to nie rób nic.
   $uniform(0, 1)$ oznacza wylosowanie liczby z przedziału $[0, 1]$ zgodnie z rozkładem jednostajnym ciągłym,
   iv) Zaktualizuj aktualną temperaturę: $T \leftarrow \delta(T)$.
4. Zwróć aktualne rozwiązanie x_c.

2.2. Przykłady

Z racji tego, że w powyższym opisie symulowanego wyżarzania pojawia się dużo funkcji/operatorów, przedstawione tu zostaną proste ich przykłady.

generowanie rozwiązania początkowego x_0 – losowe rozwiązanie dopuszczalne. Dla TSP może być to losowy cykl Hamiltona w grafie.

definicja sąsiedztwa $\mathcal{N}(x)$ – Dla pewnej funkcji $h : \mathbb{R} \to \mathbb{R}$ za sąsiedztwo x-a można przyjąć punkty oddalone od niego o pewnien ϵ. Dla TSP musi być to zbiór rozwiązań dopuszczalnych innych od x. Zatem mogą być to cykle Hamiltona w tym samym grafie będące np. permutacją x-a lub x-em z dwoma miastami zamienionymi kolejnością (1 inwersja).

sposób ochładzania δ – Może być to przykładowo jedna z funkcji (dla wszystkich funkcji β oznacza współczynnik ochładzania, i jest numerem aktualnie wykonywanej iteracji):

$$\delta_1(T, i) = T - i\beta, \text{(liniowe)}$$

$$\delta_2(T) = (1 - \beta) \cdot T$$

$\delta_3(i) = \frac{c}{\log(1+i)}$, gdzie c jest pewną odpowiednio dobraną do problemu stałą.

acceptance probability P – Może to być dowolna funkcja prawdopodobieństwa wpisująca się w ideę SA. Tzn. wraz ze spadkiem temperatury spada prawdopodobieństwo zaakceptowania gorszych rozwiązań. Przykładowa funkcja P to np. $P(x) = exp(\frac{f(x_c)-f(x)}{T})$ (T jest aktualną temperaturą).

3. Symulowane wyżarzanie z zachłannym szukaniem

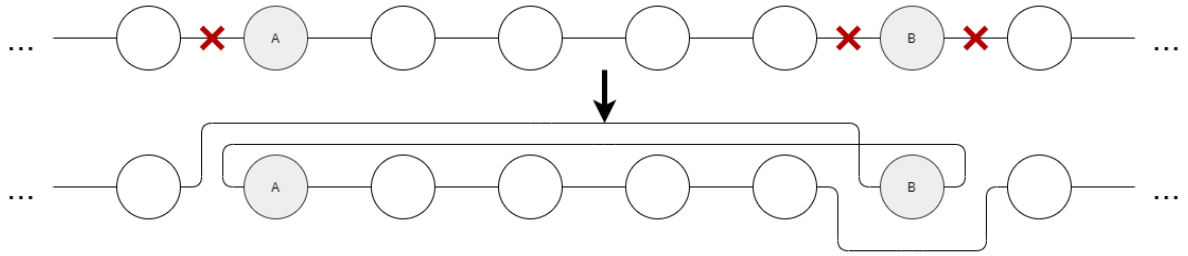
3.1. Wprowadzenie

Jest to pierwsze proponowane rozwiązanie TSP. Algorytm symulowanego wyżarzania z zachłannym szukaniem (z ang. simulated annealing algorithm with greedy search) polega na modyfikacji standardowego algorytmu symulowanego wyżarzania poprzez dodanie poszukiwania zachłannego na pewnym jego etapie. [1]. Problem sformułowany jest dokładnie tak jak w punkcie 1.1 i 1.2. Rozważany tu jest graf nieskierowany, oznacza to, że TSP jest symetryczny, czyli $d_{i,j} = d_{j,i}$, gdzie $d_{i,j}$ jest odległością z miasta i do miasta j. Dane takie jak definicja sąsiedztwa, acceptance probability (P) czy parametry związane z temperaturą lub jej zmianą zostaną podane niżej w odpowiednich miejscach.

3.2. Definicja sąsiedztwa

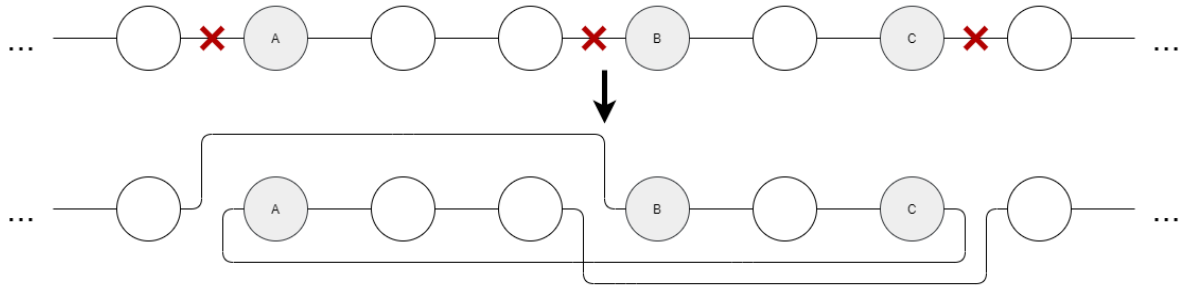
W tej metodzie sądziedztwo jest zdefiniowane w nastepujący sposób: $\hat{c}$ jest sąsiadem c, jeśli $\hat{c}$ jest mutacją c. Mutacja to zdefiniowana w pewnien sposób operacja na ciągu c (np. zamiana kolejności kilku miast). W tej metodzie mutacje będą 3 [1].

VI: Wstawienie wierzchołka (vertex insert mutation) (Rysunek poniżej):



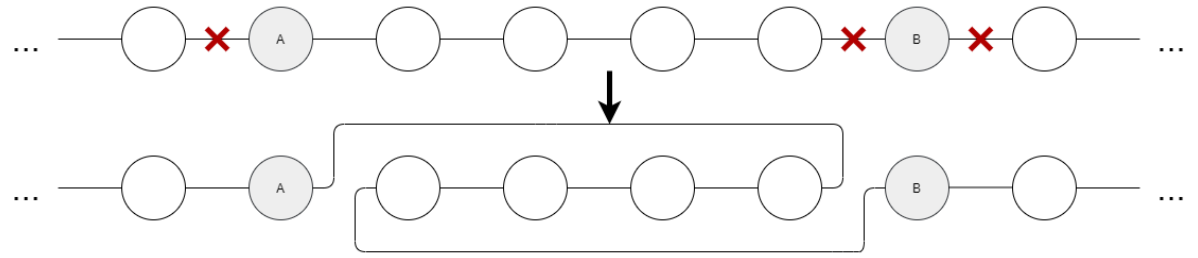
Rysunek 3: Mutacja - wstawienie wierzchołka

BI: Wstawienie bloku (block insert mutation) (Rysunek poniżej):



Rysunek 4: Mutacja - wstawienie bloku

BR: Odwrócenie bloku (block reverse mutation) (Rysunek poniżej):



Rysunek 5: Mutacja - odwrócenie bloku

Tutaj [1] (Table 1.) można znaleźć tabelę porównującą zuzycie procesora przez każda z tych metod oraz ich kombinacje. Najszybszą metodą jest BR, daje ona też najlepsze rozwiązanie. Najwolniejszą metodą, która też jest najgorsza pod względem jakości rozwiązania jest BI. Najlepsze rezulataty jeśli chodzi o jakość rozwiązania jak i szybkość uzyskano łącząc wszystkie 3 mutacje. Połączenie ich polega na braniu konkretnej mutacji z pewnym prawdopodobieństwem. Proponowane prawdopodobieństwa wynoszą: 10% dla VI, 1% dla BI, 89% dla BR.
Zapisując formalniej definicję sąsiedztwa otrzymujemy:

$$\mathcal{N}(x) = \{x\prime : x\prime \text{ jest mutacją } x, \text{według dowolnej spośród mutacji: VI, BI, BR}\}$$

3.3. Opis algorytmu

Standardowy algorytm symulowanego wyżarzania potrafi znaleźć lepsze rozwiązanie (o ile istnieje) kosztem czasu. Ideą proponowanej metody jest przyspieszenie jego zbieżności poprzez dodanie wyszukiwania zachłannego. Koncept zachłannego wyszukiwania polega tu, na wprowadzeniu dodatkowych współczynników algorymtu: t_{greedy} – mówiący ile maksymalnie prób polepszenia rozwiązania (sprawdzania sąsiadów) powinno być wykonanych przed zmianą temperatury i zachłannym wyborem najlepszego z tych t_{greedy} sąsiadów, i podjęciu decyzji o zaakceptowaniu gorszego rozwiązania. Jaśniej jest to opisane poniżej.

Stare rozwiązanie c_{old} jest zastępowane przez swojego sąsiada $c_{neighbour1}$ jeśli $F(c_{neighbour1}) < F(c_{old})$, gdzie funkcja F jest optymalizowaną funkcją (patrz punkt 1.2 – matematyczne sformułowanie TSP). Po tej operacji wykonywany jest następny krok algorytmu. W przeciwnym wypadku, czyli jeśli warunek $F(c_{neighbour1}) < F(c_{old})$ okazał się nieprawdziwy, to generowany jest kolejny sąsiad $c_{neighbour2}$ oraz sprawdzany jest analogiczny warunek do tego co wyżej tj. $F(c_{neighbour2}) < F(c_{old})$. Jeśli jest on prawdziwy, to c_{old} jest zastępowane przez swojego sąsiada $c_{neighbour2}$ i następuje przejście do kolejnego kroku algorytmu, jeśli nie to generowany jest kolejny sąsiąd i procedura się powtarza. W momencie kiedy zostanie wygenerowany t_{greedy}-ty sąsiad (wygenerowano w sumie t_{greedy} sąsiadów i żaden z nich nie okazał się lepszy od c_{old}), stare rozwiązanie c_{old} jest zastępowane przez najlepsze rozwiązanie t_{best} spośród t_{greedy} wygenerowanych przed chwilą sąsiadów z prawdopodobieństwem P.

$$c_{best} : F(c_{best}) = min\{F(c_{neighbour1}), F(c_{neighbour2}), \ldots, F(c_{neighbour\_t_{greedy}})\} \tag{1}$$

Proponowany wzór na prawdopodobiestwo P poniżej.

$$P(c_{best}) = \exp(-\frac{F(c_{best}) - F(c_{old})}{t_{current}} \cdot \frac{10n}{Opt}) \tag{2}$$

Parametr $t_{current}$ to aktualna temperatura, n to liczba miast, Opt to optymalna długość cyklu.
Metoda zmiany temperatury jest dana wzorem poniżej.

$$\delta(t_{current}) = t_{current} * \beta \tag{3}$$

Gdzie β to współczynnik ochładzania.

Metoda:

1. Za aktualne rozwiązanie x_c podstaw x_0 – rozwiązanie początkowe: $x_c \leftarrow x_0$,
2. Za aktualną temperaturę T podstaw T_0: $T \leftarrow T_0$,
3. Za zmienną G podstaw 0: $G \leftarrow 0$, będzie to zmienna służąca do liczenia ile już sąsiadów zostało sprawdzonych.
4. Dopóki $T > 0$ wykonuj kroki:
   i) Wybierz mutację $\mathcal{M}$ spośród: VI, BI, BR zgodnie z prawdopodobieństwami ich użycia.
   ii) Weź nowego sąsiada aktualnego rozwiązania. $x_{n_1} \leftarrow \mathcal{M}(x_c)$
   Wybrany x_{n_1} jest teraz kandydatem na nowe aktualne rozwiązanie,
   iii) Sprawdź czy $\triangle F = F(x_{n_1}) - F(x_c) \leqslant 0$. Jeśli tak to zaakceptuj kandydata x_{n_1}: $x_c \leftarrow x_{n_1}$, następnie przejdź do punktu **vii)**. Jeśli warunek nie był spełniony, to: $G \leftarrow G + 1$.
   iv) Jeśli nieprawda, że $G \geqslant t_{greedy}$, to przejdź do **i)** (nie zmieniając temperatury). Jeśli ten warunek jest prawdziwy, to wybierz c_{best} zgodnie z formułą **1**, $x' \leftarrow c_{best}$.
   v) Oblicz $p \leftarrow P(x', x_c, T)$ – prawdopodobieństwo zaakceptowania kandydata x',
   P jest dane formułą 2,
   vi) Jeśli $p \geqslant uniform(0, 1)$ to podstaw: $x_c \leftarrow x'$. Jeśli nie, to nie rób nic.
   vii) Zaktualizuj aktualną temperaturę: $T \leftarrow \delta(T)$ oraz $G \leftarrow 0$. δ jest zadana formuła 3.
5. Zwróć aktualne rozwiązanie x_c.

3.4. Dobór parametrów

Parametry takie jak: t_{greedy}, t_{start} (temperatura początkowa), β, P będą zależały od *skali problemu*. Szybkie ochładzanie będzie działać dobrze, dla małej ilości miast. Dla instancji TSP, gdzie długość optymalnej ścieżki będzie długa odpowiednie będą wysokie temperatury początkowe. Proponowane współczynniki (4) są dla problemu komiwojażera gdzie liczba miast waha się od 51 do 85900.

$$\begin{aligned} \beta &= \frac{(\alpha \cdot n^{0.5} - 1.0)}{\alpha \cdot n^{0.5}} \\ t_{greedy} &= \beta \cdot n \\ P(c_{best}) &= \exp(-\frac{F(c_{best}) - F(c_{old})}{t_{current}} \cdot \frac{10n}{Opt}) \end{aligned} \tag{4}$$

4. Proponowane rozwiązanie problemu II

5. Wnioski

- męczące to bardzo

Bibliografia

[1] Xiutang Geng, Zhihua Chen, Wei Yang, Deqian Shi, Kai Zhao. *Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search*, 2011

[2] Walid Mahdi, Seyyid Ahmed Medjahed, Mohammed Ouali. *Performance Analysis of Simulated Annealing Cooling Schedules in the Context of Dense Image Matching*, 2017
`http://www.scielo.org.mx/pdf/cys/v21n3/1405-5546-cys-21-03-00493.pdf`