



Politechnika
Wrocławska

Algorytmy sztucznej inteligencji

Tabu Search



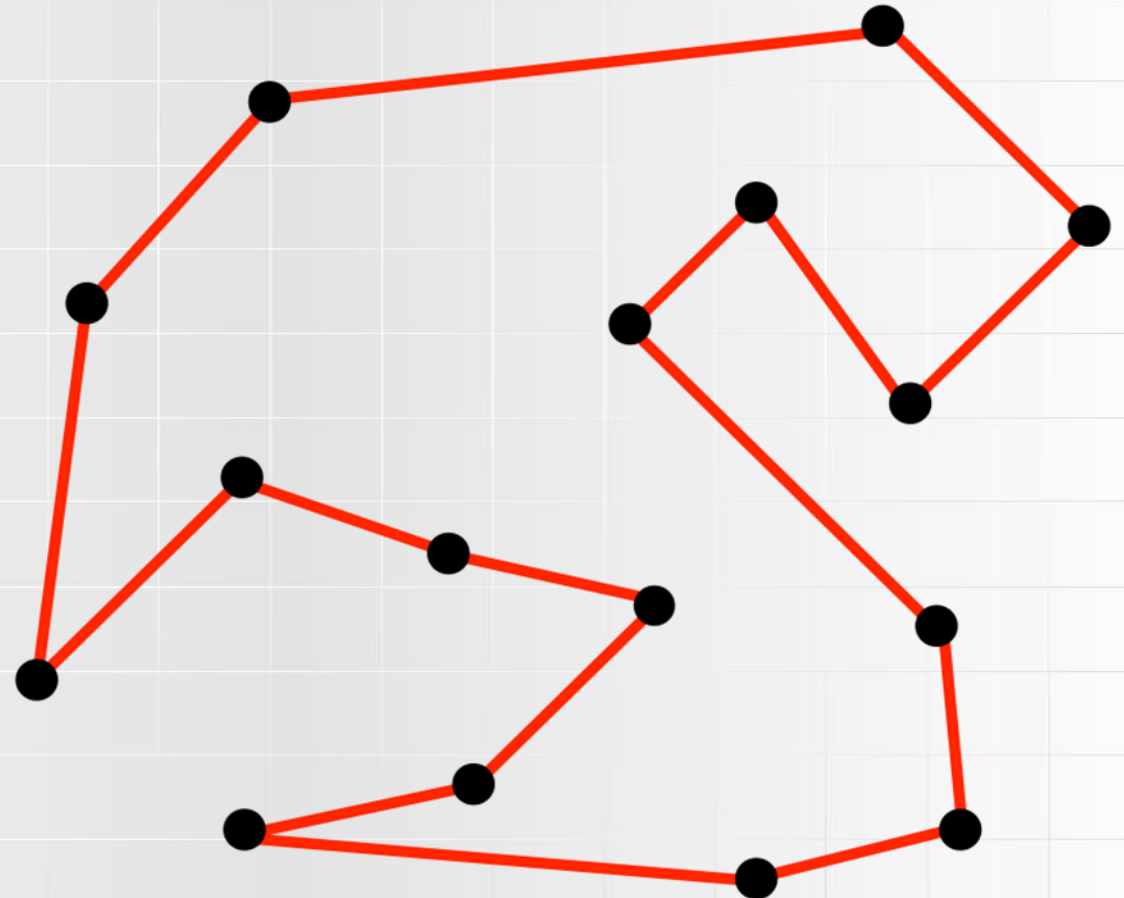
HR EXCELLENCE IN RESEARCH



Michał Bagiński
Szymon Wąchała
Filip Wolski

TSP - problem komiwojażera

- Dane jest n miast, które komiwojażer ma odwiedzić, oraz odległość pomiędzy każdą parą miast.
- Celem jest znalezienie najkrótszej drogi łączącej wszystkie miasta, zaczynającej się i kończącej się w określonym punkcie



Tabu Search

- algorytm metaheurystyczny,
- umożliwia znalezienie rozwiązania bliskiego optymalnemu,
- polega na przeglądaniu tylko części z przestrzeni rozwiązań,
- algorytm posiada tablicę tabu - tablicę ruchów zabronionych: rozwiązań lub ruchów,
- tabu search jest algorytmem deterministycznym.

```
pi0 - nie powinno być losowe,  
pi0 - musi być dopuszczalne - brak cyłki,  
LT = 0,  
bestPi = pi0,  
pi = pi0  
repeat:  
    - wygeneruj otoczenie N(pi) rozwiązania pi,  
    - usuń z N(pi) rozwiązanie zabronione przez LT  
      chyba, że są lepsze niż bestPi - kryterium aspiracji,  
    - oblicz funkcję celu dla każdego elementu z sąsiedztwa,  
    - jeśli najlepsze rozwiązanie w sąsiedztwie jest lepsze niż  
      najlepsze dotychczasowe to bestPi = pi,  
    - dodaj atrybuty rozwiązania do listy tabu,  
until stop criteria
```

Generowanie permutacji początkowej

Permutacja początkowa jest generowana za pomocą algorytmu zachłannego o złożoności obliczeniowej $O(n^2)$.

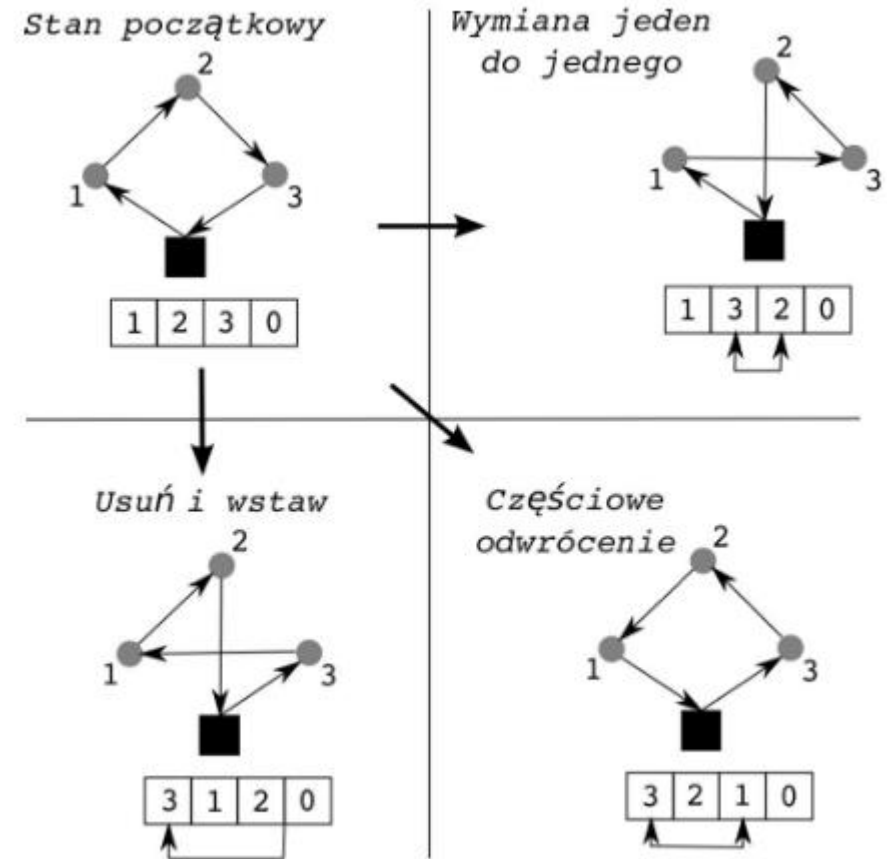
Umożliwia to już na wstępie wejść do algorytmu tabu search z rozwiązaniem bliskim optymalnemu.



Generowanie sąsiedztwa rozwiązania

Sposoby generowania sąsiedztwa:

- 1) poprzez wstawianie,
- 2) poprzez zamianę,
- 3) częściowe odwrócenie,
- 4) API – zamiana par przyległych,
- 5) NPI – zamiana par nieprzyległych.



Rys. 9.3. Przekształcenia Hisafumi Kokubugata i Hironao Kawashima

Implementacja algorytmu

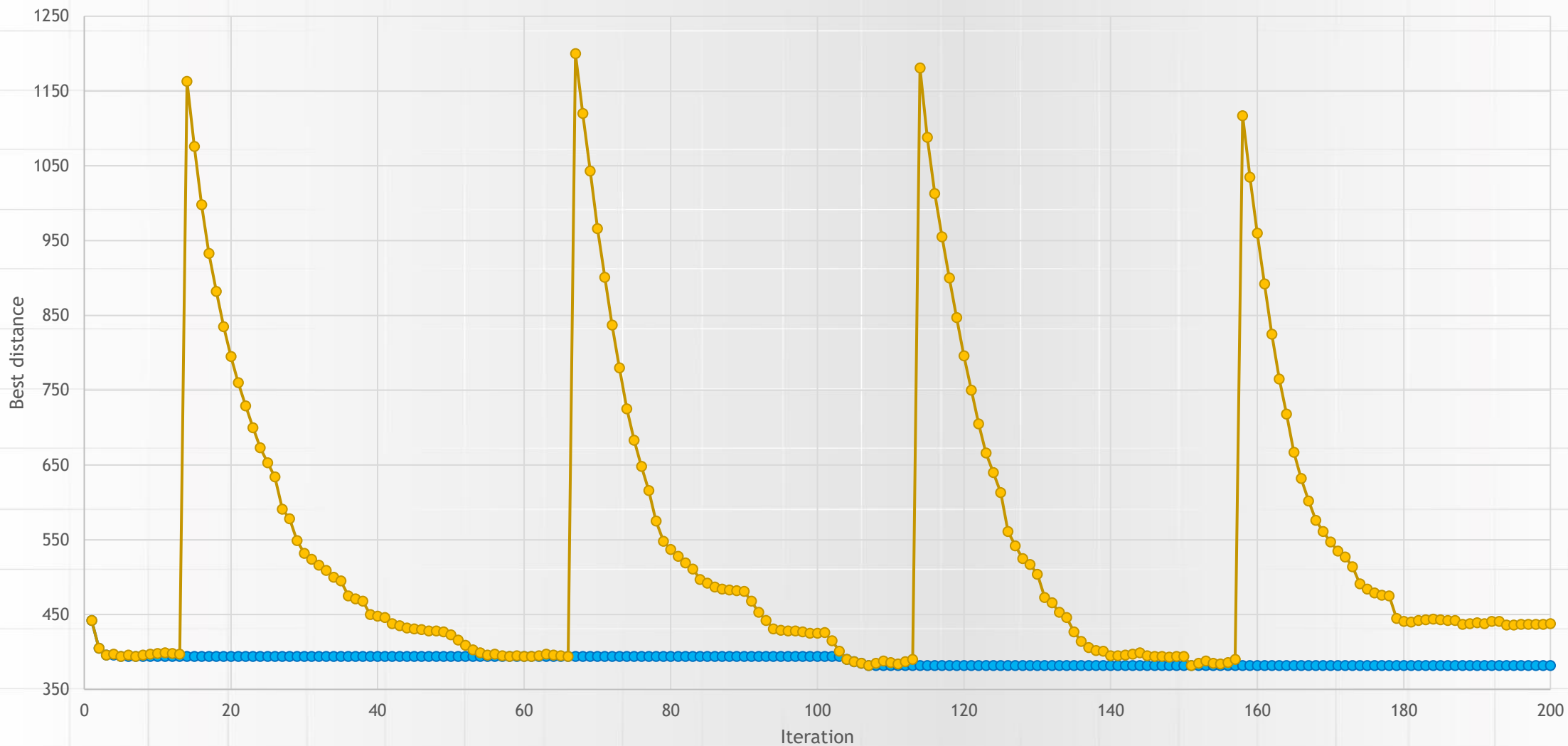
```
def execute(self, startPermutation, lenghtOfTabu, option, iterationNumber, cycleNumberMax, isReactiveTabu, reactiveInterval):
```

Przekazywane parametry:

- startPermutation – permutacja startowa,
- lenghtOfTabu - długość listy tabu,
- option - sposób generowania sąsiedztwa np. "swap", "insert",
- iterationNumber – maksymalna ilość iteracji,
- cycleNumberMax – maksymalna długość cyklu,
- isReactiveTabu - czy algorytm ma działać w trybie reaktywnym,
- reactiveInterval - ilość iteracji po wygenerowaniu rozwiązania losowego.

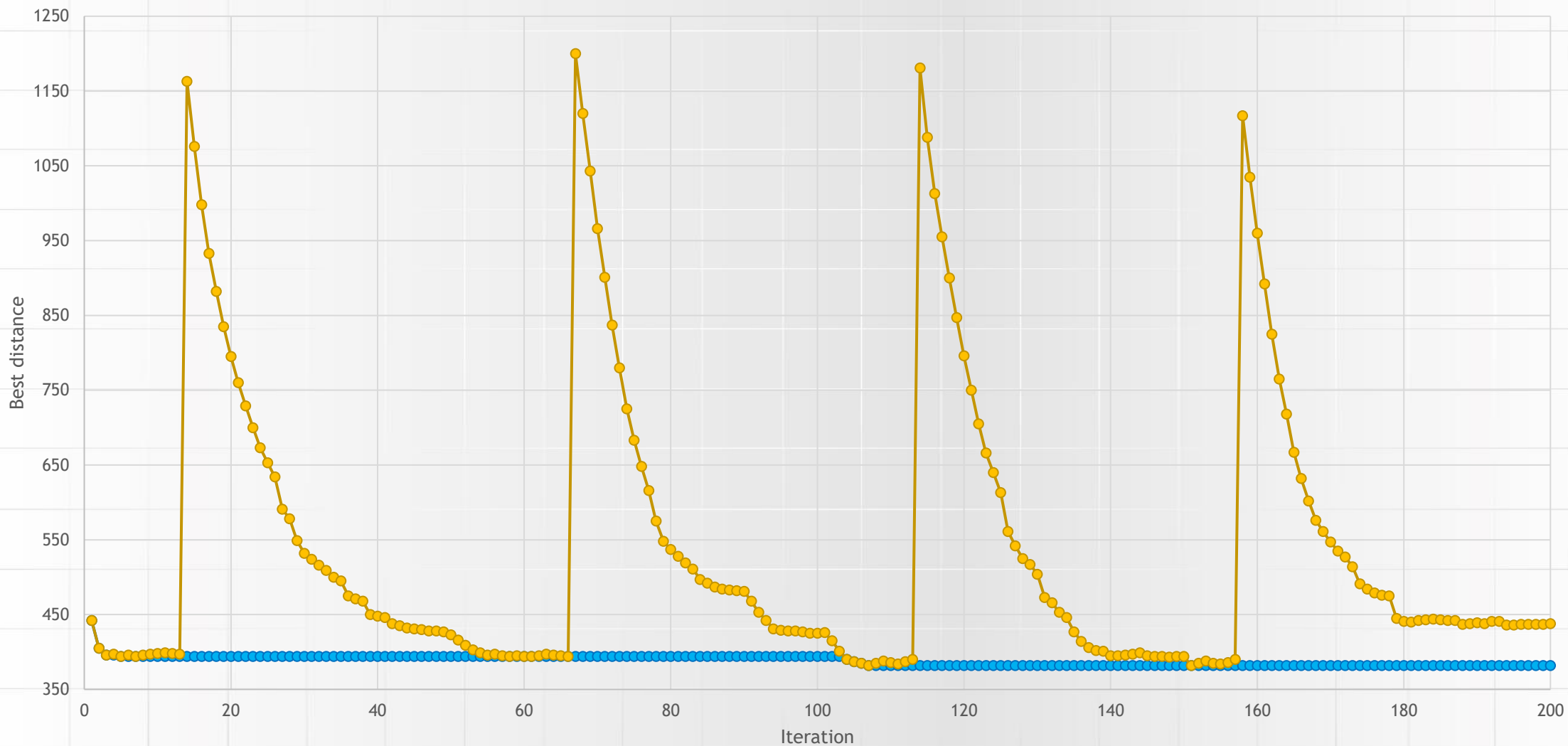
Przeszukiwanie przestrzeni rozwiązań

(rozpoczynając od zachłannie wygenerowanej permutacji)



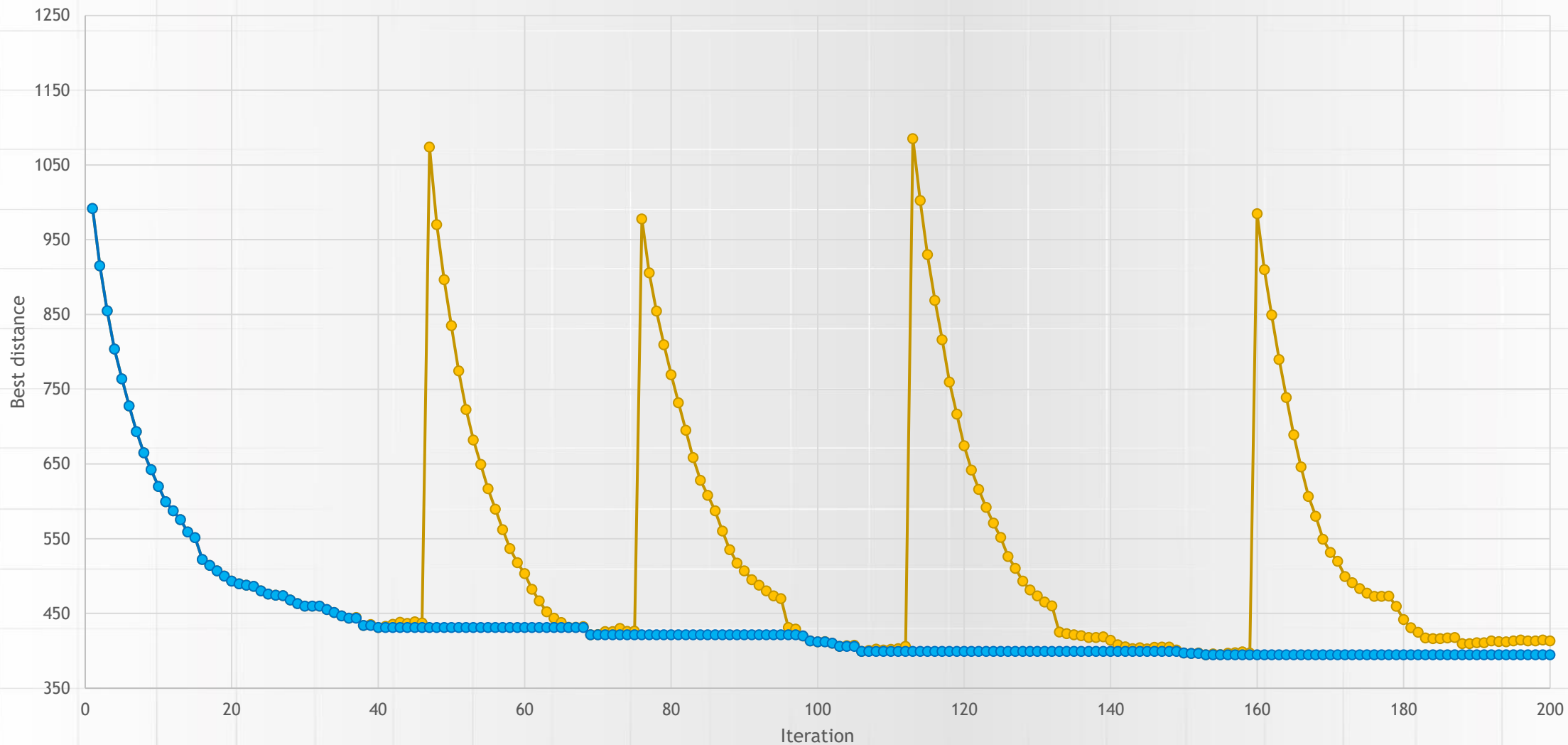
Przeszukiwanie przestrzeni rozwiązań

(rozpoczynając od zachłannie wygenerowanej permutacji)



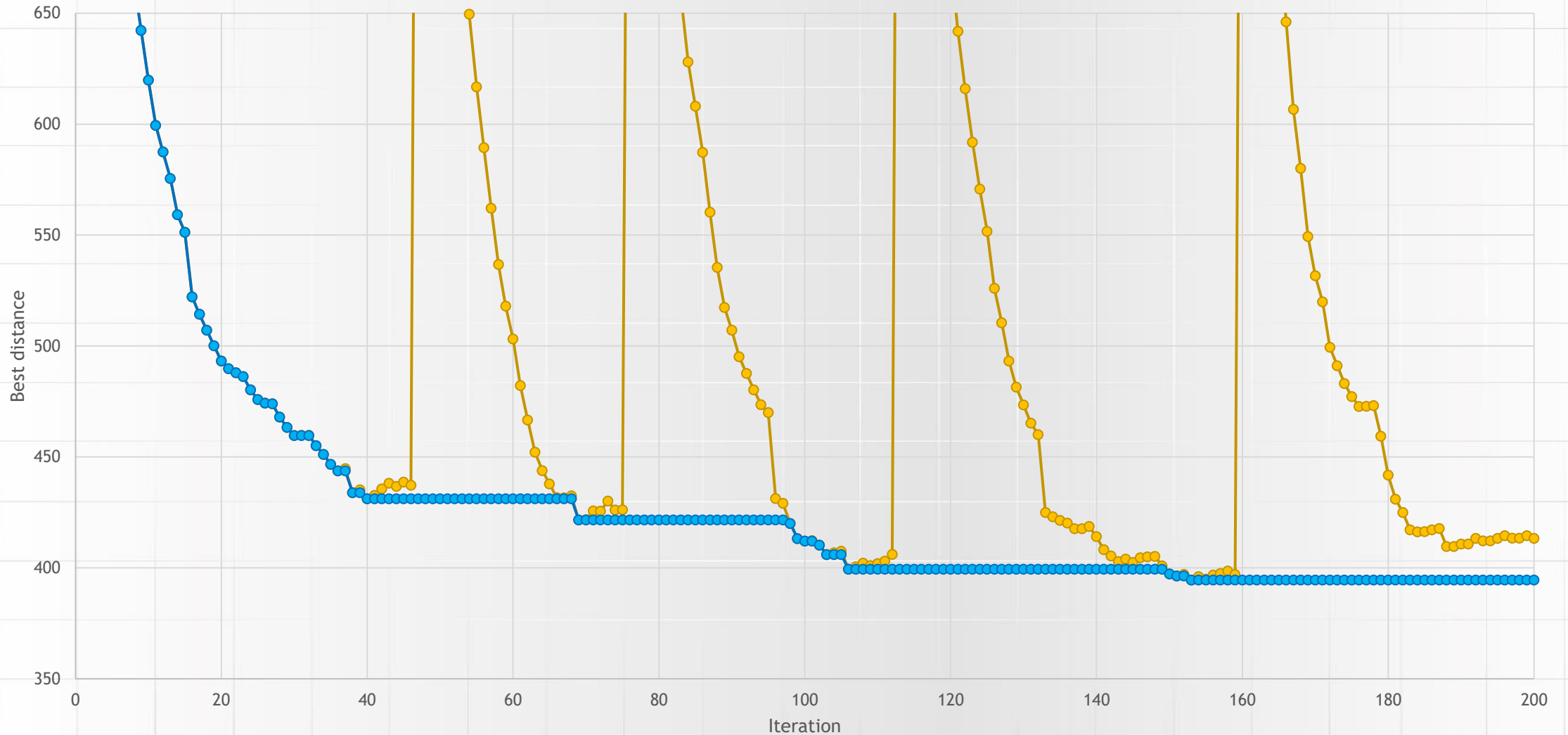
Przeszukiwanie przestrzeni rozwiązań

(rozpoczynając od zachłannie wygenerowanej permutacji)

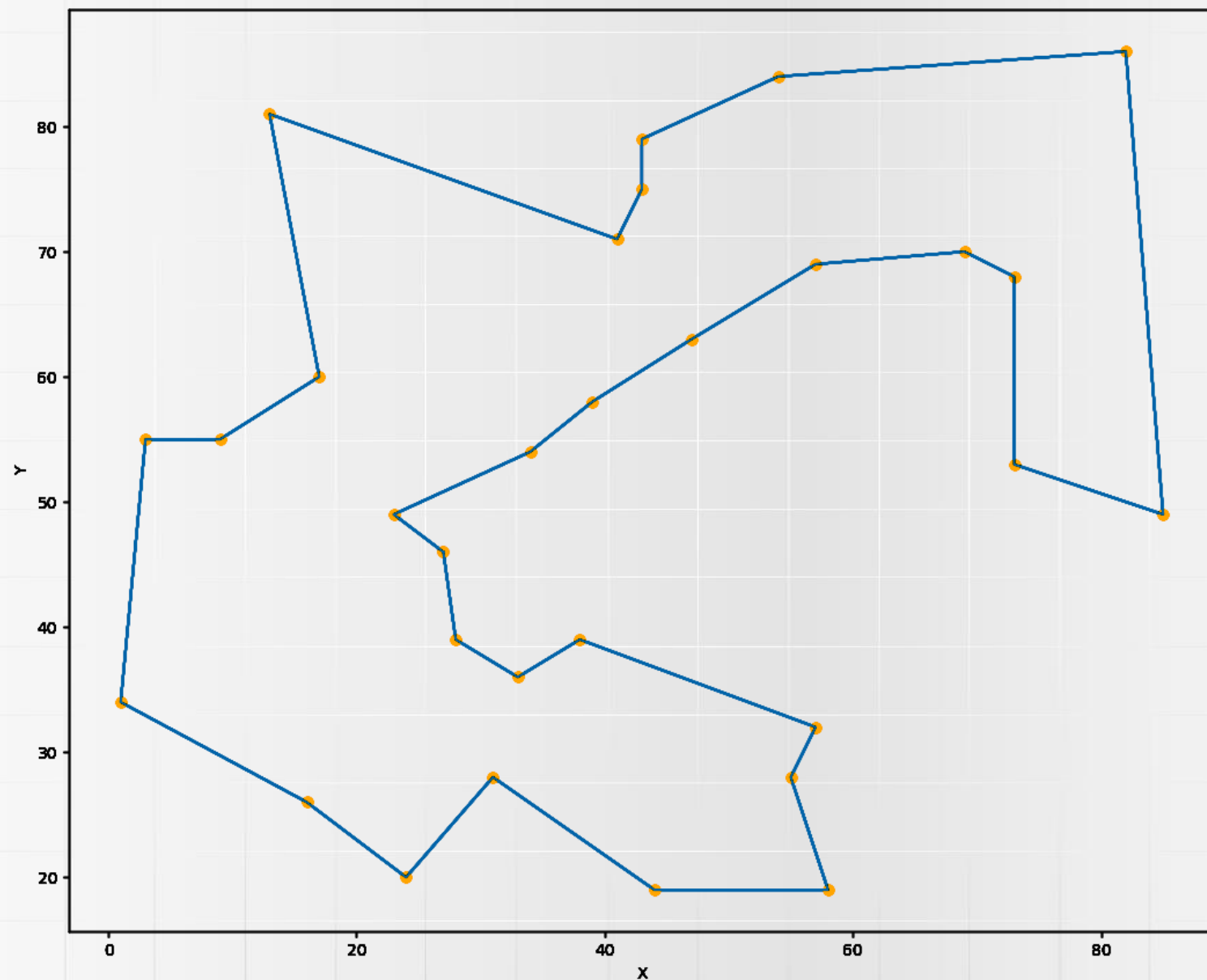


Przeszukiwanie przestrzeni rozwiązań

(rozpoczynając od zachłannie wygenerowanej permutacji)



Uzyskana trasa



Porównanie Tabu Search z algorytmem 2-opt oraz SA

Ilość miast	Długość trasy	Algorytm
15	354,85	Tabu Search
30	396,75	
51	456,13	
15	391,81	2-opt
30	418,95	
51	468,02	
15	354,85	Symulowane wyżarzanie
30	421,65	
51	477,71	



Politechnika
Wrocławska

Dziękujemy za uwagę

