Wydział Zarządzania AGH
Katedra Informatyki Biznesowej i Inżynierii Zarządzania

Heurystyki i metaheurystyki w podejmowaniu decyzji

SWD

Heurystyki, metaheurystyki

- Wprowadzenie, podstawowe pojęcia
- Proste metody przeszukiwania
- Metaheurystyki inspirowane naturą
- Symulowane wyżarzanie
- Metaheurystyka ILS
- Tabu search
- Metody badania algorytmów

Wprowadzenie, podstawowe pojęcia

- Inteligencja obliczeniowa (Computational Intelligence, CI)
  - Dziedzina nauki zajmująca się rozwiązywaniem problemów, które nie są efektywnie algorytmizowane.
  - Computational intelligence to inteligencja zbudowana przy użyciu systemu komputerowego.
  - CI z reguły próbuje naśladować inteligencję człowieka, aczkolwiek nie jest to ani ograniczenie, ani wymaganie.
  - Synonim inteligencji maszynowej.
  - Synonim sztucznej inteligencji.

IEEE Computational Intelligence Society
Nature-Inspired Problem Solving

Wprowadzenie, podstawowe pojęcia

- Organizacje (przede wszystkim gospodarcze i rządowe) są głównie zainteresowane dwoma zadaniami:
  1. przewidywaniem tego, co się ma wydarzyć (predykcja),
  2. podejmowaniem najlepszych decyzji w warunkach ryzyka i niepewności (optymalizacja).
- Zadaniem CI jest dostarczyć narzędzia do modelowania, symulacji i optymalizacji zaspokajające te potrzeby.

Klasyczne metody prognozowania, i:

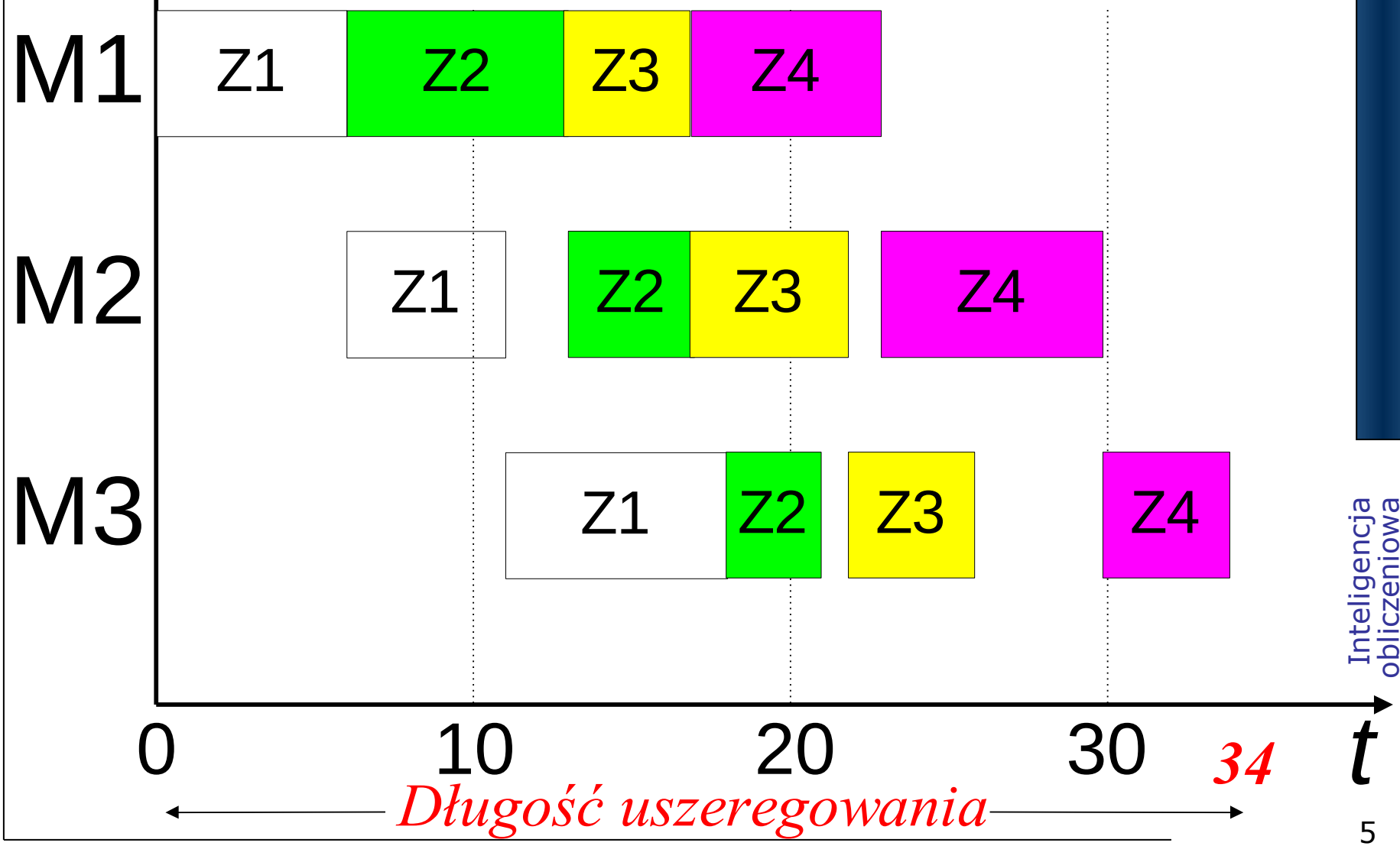
1. Sieci neuronowe,
2. Systemy rozmyte,
3. Programowanie genetyczne,
4. Systemy agentowe,
5. Techniki DM.

Klasyczne metody OR, i:

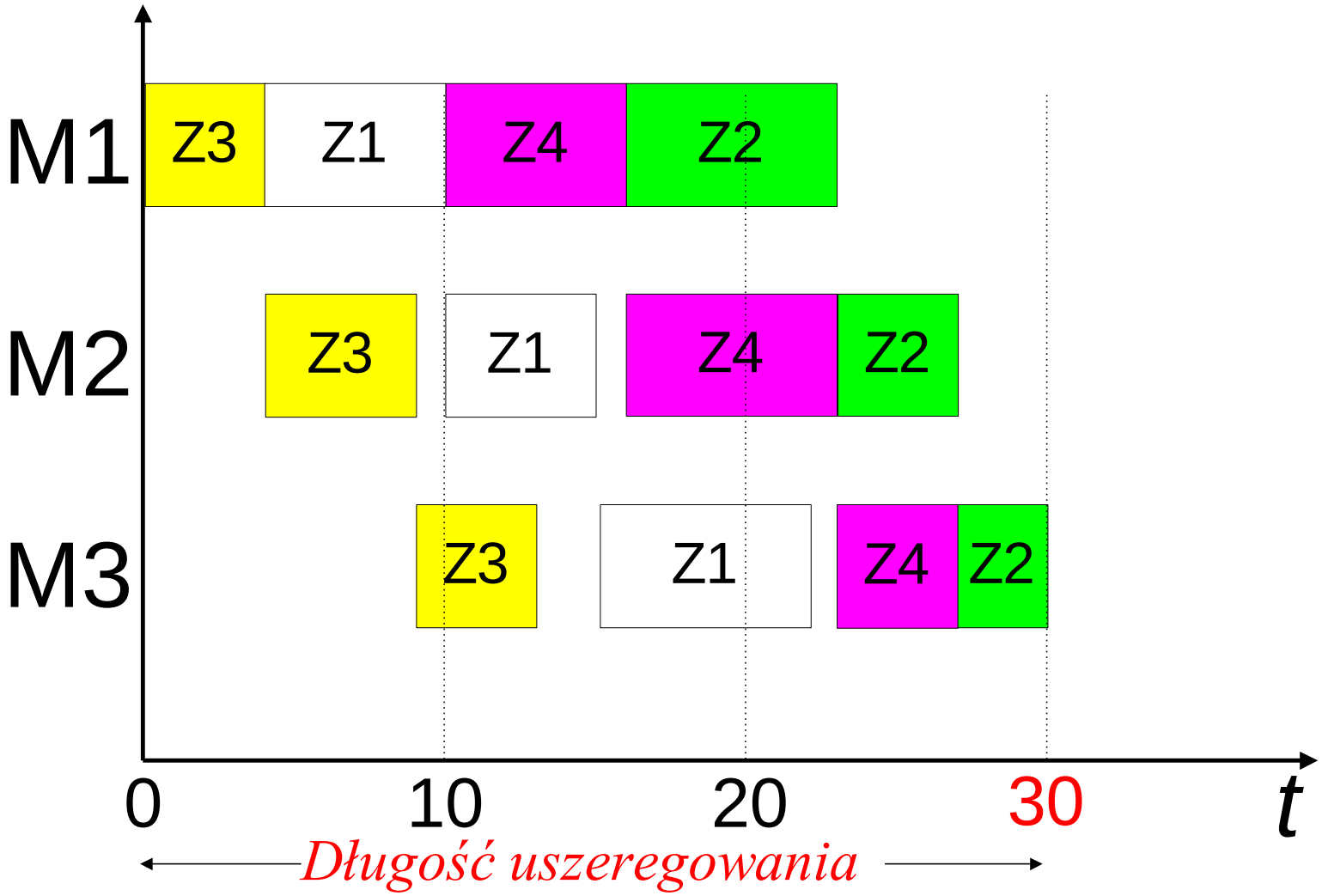
1. Algorytmy ewolucyjne,
2. Inteligencja roju,
3. Symulowane wyżarzanie,
4. Tabu search,
5. Algorytmy mrówkowe.

Źródło: Z. Michalewicz, Adaptive business intelligence.

Wprowadzenie, podstawowe pojęcia



Wprowadzenie, podstawowe pojęcia



Wprowadzenie, podstawowe pojęcia

- GM sprzedawał rocznie 1,2 mln poleasingowych samochodów na różnych aukcjach. Codziennie decydowano, gdzie dostarczyć 4000-7000 aut.

Na problem wpływają:

- popyt,
- stan samochodów,
- trasy transportowe,
- koszt kapitału,
- ryzyko handlowe,
- efekt skali.

Źródło: Z. Michalewicz, Adaptive business intelligence.

Heurystyka, metaheurystyka

- *Heuriskein* – szukam, sztuka znajdowania rozwiązań problemów.
- Heurystyka:
  - niepewność wyniku,
  - niekompletność dostępnej wiedzy,
  - poprawianie rozwiązania,
  - doświadczenie, intuicja.
- W inteligencji obliczeniowej termin *heurystyka* oznacza właściwie przeciwieństwo metody dającej optymalne rozwiązanie. Początek – lata 60.

Heurystyka, metaheurystyka

Heurystyka jest specjalizowaną metodą rozwiązywania konkretnego problemu, która znajduje dobre rozwiązania przy akceptowalnych nakładach obliczeniowych, ale bez gwarancji osiągnięcia optymalności celu, czy nawet - w wielu przypadkach – jak blisko optymalnego jest otrzymane rozwiązanie.

Metaheurystyka - ogólna metoda służąca za szkielet do konstrukcji heurystyki rozwiązującej dowolny problem, który można opisać za pomocą pewnych definiowanych przez tę metodę pojęć.
Metody takie nie służą do rozwiązywania konkretnych problemów, a jedynie podają sposób na utworzenie odpowiedniego algorytmu heurystycznego.

Cechy metaheurystyk

- metody określające sposób przeszukiwania przestrzeni dopuszczalnych rozwiązań problemu
- celem działania jest efektywne przeszukiwanie przestrzeni
  - znajdowanie dobrych rozwiązań w określonym regionie (**eksploatacja**)
  - przeglądanie możliwie najszerszego obszaru przestrzeni problemu (**eksploracja**)
- metody przybliżone i przeważnie niedeterministyczne
- stosują różne sposoby: od prostego przeszukiwania lokalnego do skomplikowanych procesów ewolucyjnych
- wykorzystują mechanizmy zapobiegające utknięciu metody w ograniczonym obszarze przestrzeni problemu
- nie są specjalizowane dla żadnego specyficznego problemu
- wykorzystują wiedzę o problemie i/lub doświadczenie zgromadzone w procesie przeszukiwania przestrzeni

Heurystyka, metaheurystyka

Argumenty za:

- Akceptowalna złożoność obliczeniowa,
- Elastyczność.

Podstawowe heurystyki:

- Losowe przeszukiwanie
- Algorytmy zachłanne

Podstawowe metaheurystyki:

- Metoda wspinaczki (hill climbing)
- Symulowane wyżarzanie
- Tabu Search
- Iterated Local Search
- Algorytmy ewolucyjne
- Algorytmy mrówkowe
- Inteligencja roju

1. Szczegółowe informacje na temat omawianych tu metod są łatwe do znalezienia.
2. Jeśli znasz inne metody, które mogą być interesujące dla uczestników tego kursu, daj znać – podyskutujemy.

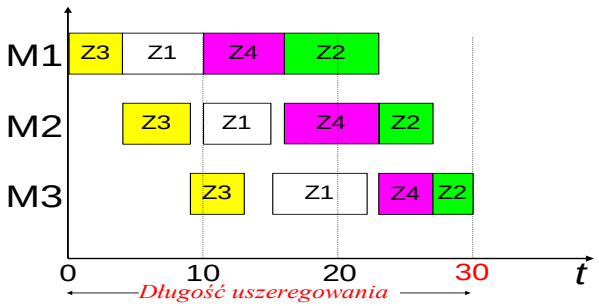
Proste algorytmy heurystyczne

- Pełny przegląd przestrzeni poszukiwań.
- Przeszukiwanie losowe
  Obecnie rzadko stosowane podejście: polega na losowym lub dokonywanym wg przyjętych reguł wyborze punktów w przestrzeni dopuszczalnych rozwiązań i obliczaniu dlań funkcji celu. Najlepsze rozwiązanie jest uznawane za rozwiązanie problemu.
- Algorytm zachłanny (greedy algorithm)
  algorytm, który w celu wyznaczenia rozwiązania w każdym kroku dokonuje zachłannego, tj. najlepiej rokującego w danym momencie wyboru **rozwiązania częściowego**. Algorytm zachłanny nie bada, jaki jest skutek danego działania w kolejnych krokach, ale dokonuje decyzji lokalnie optymalnej, kontynuując rozwiązanie podproblemu wynikającego z podjętej decyzji.
- Przykład: NN dal TSP, NEH dla FP, wydawanie reszty.

Proste metody heurystyczne

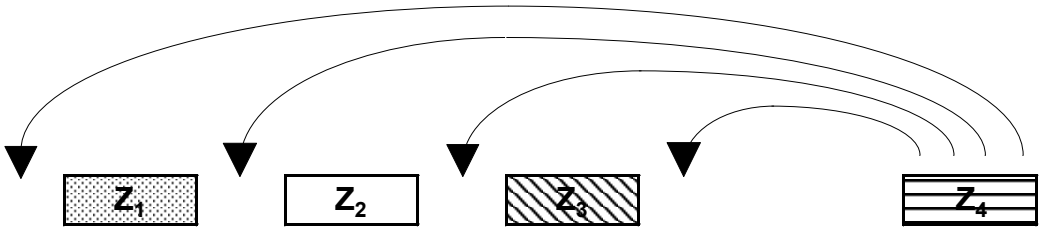
[redacted]

Proste metody heurystyczne

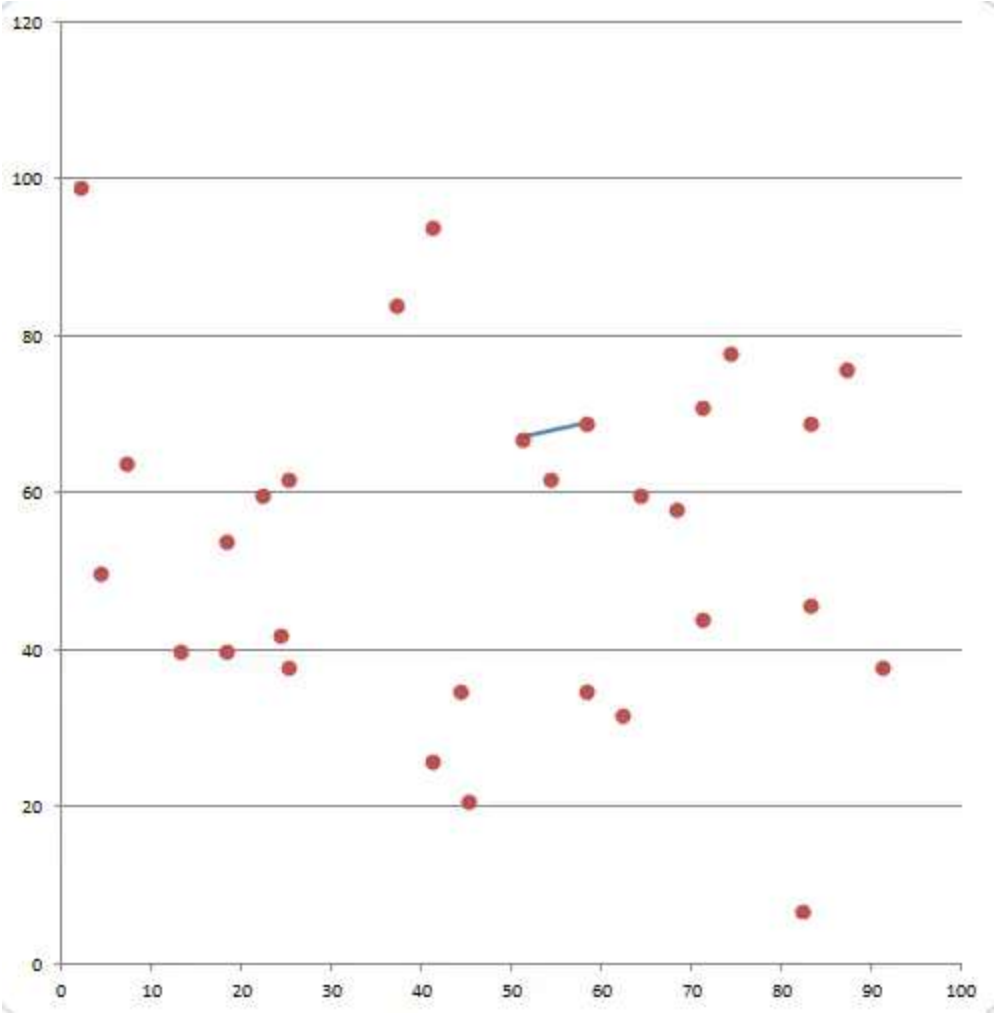


Algorytm NEH:

- uporządkuj wszystkie zadania wg malejącej sumy czasów obróbki,
- ustaw dwa pierwsze zadania tak, by otrzymać minimalną długość uszeregowania częściowego,
- dla k=3 do n:
  - umieść zadanie nr k w jednym z k możliwych miejsc (por. rysunek); jako rozwiązanie częściowe wybierz to, które daje minimalną długość uszeregowania)



Proste metody heurystyczne



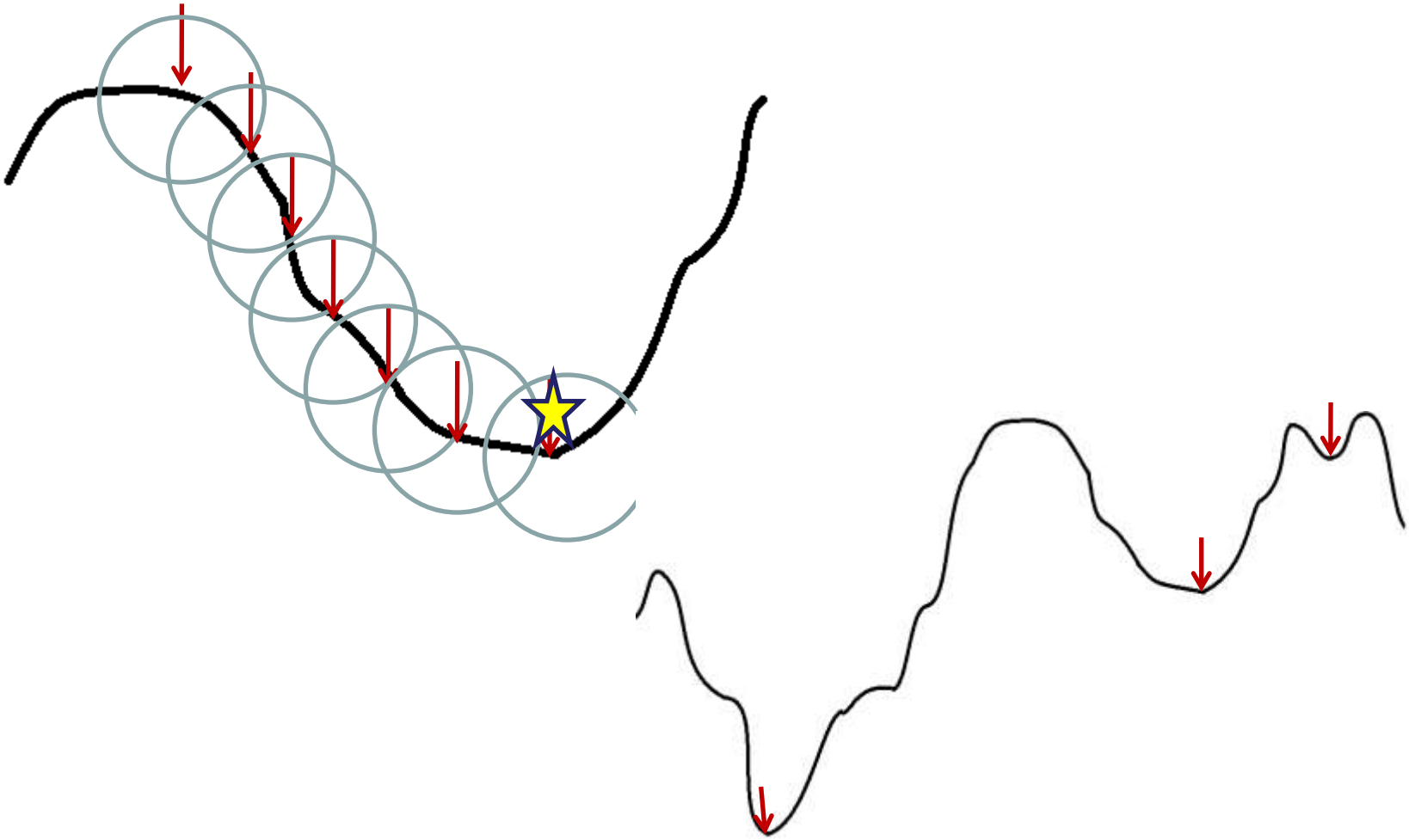
Proste metody heurystyczne

- Metoda wspinaczki (hill climbing)

1. Wybierz rozwiązanie początkowe (najczęściej losowo).
2. Przeanalizuj sąsiadów rozwiązania i określ ich jakość.
3. Wybierz najlepsze rozwiązanie jako początkowe do następnej iteracji.
4. Powtarzaj 2-4, aż znalezienie rozwiązania lepszego nie będzie możliwe.
5. Wyprowadź rozwiązanie aktualne jako rozwiązanie problemu.

Proste metody heurystyczne

Metoda wspinaczki (hill climbing)



Proste metody heurystyczne

[redacted]

Proste metody heurystyczne

[redacted]

Proste techniki heurystyczne

[redacted]

Proste techniki heurystyczne

- Przeszukiwanie iteracyjne
  Połączenie przeszukiwania losowego i wspinaczki daje ciekawą metodę przeszukiwania iteracyjnego. Po znalezieniu optimum lokalnego, procedura przeszukiwania jest wznawiana w nowym, losowo wybranym, punkcie startowym. Jest to bardzo prosta metoda i bardzo efektywna dla problemów z niewielką liczbą optimów lokalnych.
  Każde przeszukiwanie jest samodzielnym procesem wykonywanym niezależnie od pozostałych, stąd nie wykorzystuje się wiedzy o poprzednich rozwiązaniach. Tak więc przeszukiwanie jest równie prawdopodobne dla obszarów o zróżnicowanych wartościach funkcji celu.

Rodzaje metaheurystyk

Inspirowane przyrodą (naturą)

Zjawiska biologiczne i społeczne:

- Algorytmy ewolucyjne,
- Algorytmy mrówkowe,
- Sztuczne systemy immunologiczne,
- Inteligencja roju (owady, ptaki),
- Ewolucja kulturalna,
- Tabu search.

Zjawiska fizyczne i chemiczne: symulowane wyżarzanie.

Inspirowane pozaprzyrodniczo: ILS, VNS

Populacyjne vs. niepopulacyjne (pojedyncze rozwiązanie)

Symulowane wyżarzanie

Simulated annealing (SA)

Podstawy działania

Ogólnie ujmując proces odprężania (wyżarzania) polega na powolnym obniżaniu temperatury tak, by w każdej temperaturze materiał mógł osiągnąć stan równowagi. Takie postępowanie zapobiega powstawaniu wewnętrznych naprężeń i pozwala na osiągnięcie globalnego minimum energii, odpowiadające idealnemu kryształowi. Wyznaczenie stanu najniższej energii jest problemem optymalizacyjnym, stąd fizyczny proces odprężania posłużył jako wzór do opracowania nowej metody optymalizacji globalnej.

Symulowane wyżarzanie

Podstawy działania

Opis matematyczny stworzył Metropolis z zespołem w 1953 roku. Punktem wyjścia były prace M. z dziedziny termodynamiki statystycznej, gdzie obowiązuje:

$$p(E)=\exp(-dE/kt)$$

Stała Bolzmana

- Kirkpatrick z zespołem wykazali, że metoda ta może być zastosowana do problemów optymalizacyjnych, drogą analogii do procesów fizycznych.

Symulowane wyżarzanie

Podstawy działania

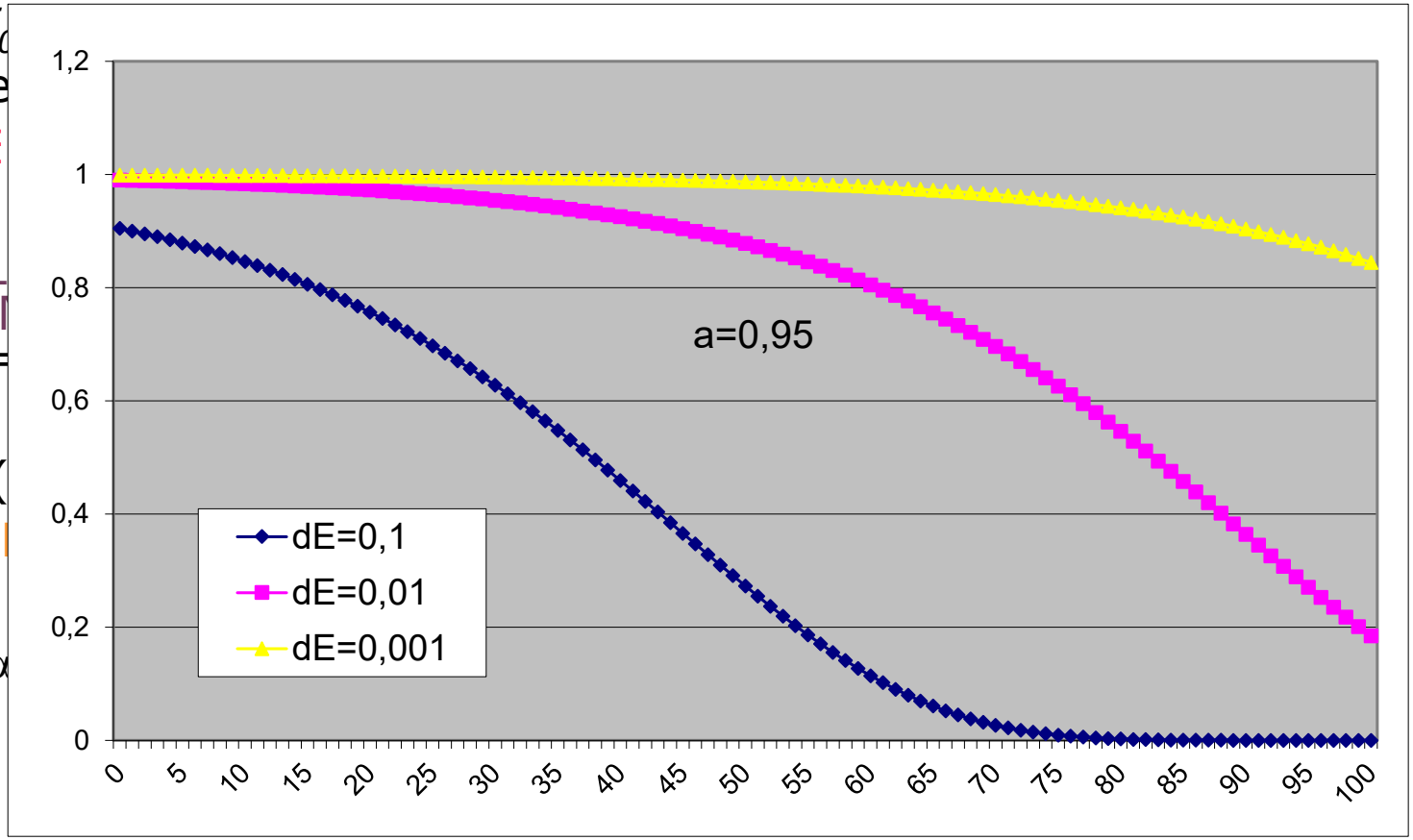
Wyżarzanie	Optymalizacja
stany systemu	dopuszczalne rozwiązania
energia	koszt
zmiana stanu	rozwiązanie sąsiednie
temperatura	parametr sterujący
stan zamrożony	rozwiązanie heurystyczne

Symulowane wyżarzanie

- SA należy do metod przeszukiwania sąsiedztwa.
- Sąsiedztwo N(x,m) rozwiązania x jest zbiorem rozwiązań, które mogą być osiągnięte z x drogą prostej operacji m, zwanej ruchem. Jeśli rozwiązanie y jest lepsze od dowolnego z jego sąsiedztwa N(y,m), wtedy y jest optimum lokalnym.
- Wymagania odnośnie sąsiedztwa:
  - niewielki rozmiar - liczba rozwiązań sąsiednich,
  - możliwość osiągnięcia każdego rozwiązania z dowolnego rozwiązania początkowego,
  - rozwiązania sąsiednie muszą być podobne.

Symulowane wyżarzanie

```
BEGIN
    R := R0 {rozwiązanie początkowe};
    T := T0
      wybie
    WHILE
    BEGIN
      WHIL
      BEGI
       R' :=
        dE:
        IF (
       ELSI
      END;
      T:= α
    END;
END.
```



Symulowane wyżarzanie

Warunkiem poprawnego działania przedstawionej metody jest prawidłowy dobór dwóch grup czynników:

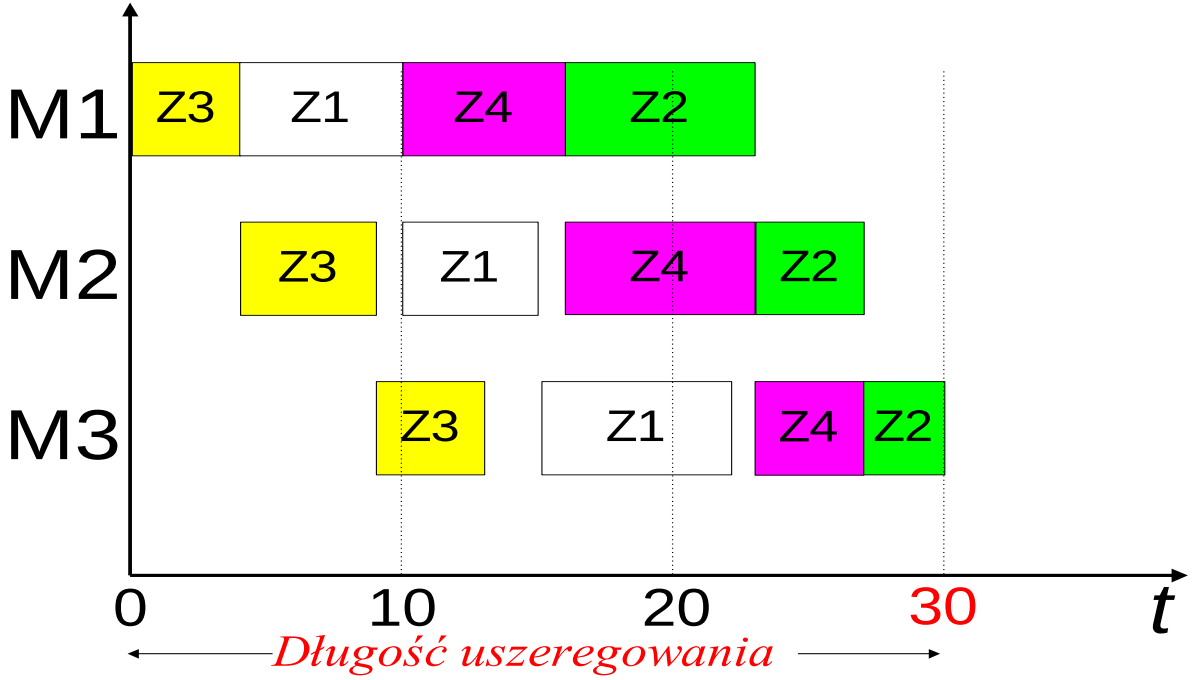
- warunek zatrzymania obliczeń,
- prawdopodobieństwo akceptacji "gorszego" rozwiązania,
- liczba generowanych rozwiązań dla danej temperatury,
- sposób generowania sąsiednich rozwiązań,
- postać funkcji kosztowej.

Kluczowym czynnikiem jest szybkość (sposób) redukcji temperatury. Najczęściej stosuje się dwa sposoby:

- redukcja geometryczna
  $\alpha(T)=a*T, \quad a<1$
  w danej temperaturze testowanych jest wiele rozwiązań,
- powolny spadek
  $\alpha(T)=T/(1+b*T), \quad b<<1$
  w danej temperaturze testowane jest jedno rozwiązanie.

Symulowane wyżarzanie - przykład

- Udane implementacje schematu Metropolisa dla problemu szeregowania zadań w systemie przepływowym podali Ogbu i Smith, Osman i Pots oraz Gangadharan i Rajendran.



Symulowane wyżarzanie - przykład

- Ogbu i Smith przyjęli, że działanie heurystyki przebiega w *LE* = 15 etapach, liczba sprawdzanych rozwiązań na każdym etapie jest funkcją liczby zadań *n* i równa się $5*(n-1)^2$ dla $n<15$ oraz $7*(n-1)^2$ dla $n>=15$, natomiast prawdopodobieństwo akceptacji *AP(k)* dla $k=1,\ldots,LE$ jest dane wzorem:

Prawd. bazowe = 0,2

$$AP(k)=\begin{cases} AP(1)\bullet(a)^{k-1} & \text{jeżeli } du(R')>du(R) \\ 1 & \text{jeżeli } du(R')\leq du(R) \end{cases}$$

Czynnik redu kujący = 0,75

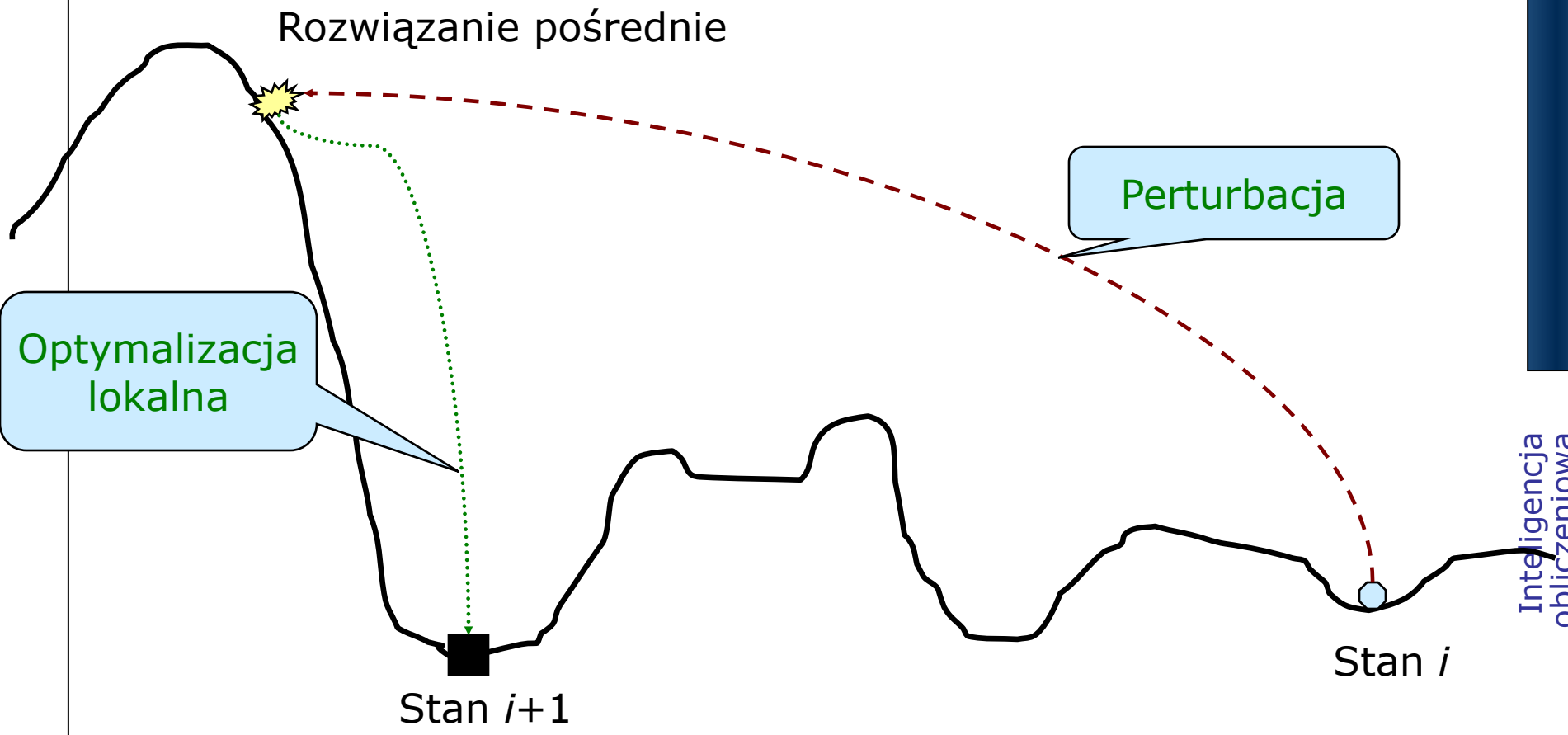
Symulowane wyżarzanie - przykład

- Nowe, sąsiednie rozwiązania generowane są przy użyciu schematu nazywanego *insertion*. Polega on na tym, że po wylosowaniu dwóch liczb *i, j* (*i*<>*j*), zadanie nr *i* jest wstawiane na pozycję nr *j* sekwencji wejściowej.

	1	2	3	4	5
i=2, j=4	1	3	4	2	5
i=4, j=2	1	4	2	3	5

- Po wygenerowaniu na danym etapie wszystkich sekwencji wybierana jest ostatnia zaakceptowana jako bazowa dla następnego etapu.

Metaheurystyka ILS (CLO)



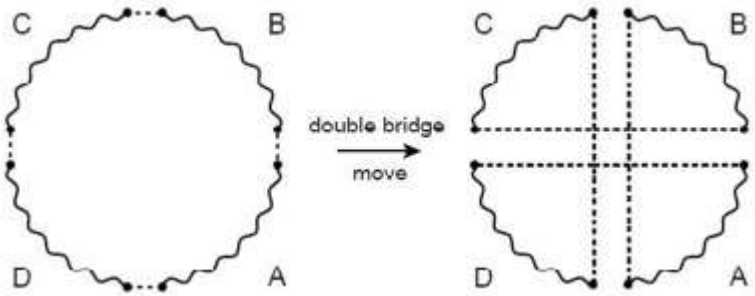
Metaheurystyka ILS (CLO)

```
s(0) = GenerateInitialSolution
s* = LocalSearch(s_0)
REPEAT
    s' = Perturbation(s*,history)
    s*' = LocalSearch(s')
    s* = AcceptanceCriterion(s*,s*',history)
UNTIL termination condition met
```

Metaheurystyka ILS (CLO)

Klasyczne podejście *ILS* do *TSP* składa się z następujących elementów:

- sąsiedztwo: *zamiana* dwóch miast;
- perturbacja: *double-bridge* move;



- kryterium akceptacji: “Czy nastąpiła poprawa najlepszego rozwiązania?”

$$s'^* < s^*$$

Tabu search

Tabu (ang. Taboo) (z polinezyjskiego tapu)

1) w religiach ludów pierwotnych zakaz podejmowania różnych działań w stosunku do osób, miejsc, zwierząt, rzeczy bądź stanów rzeczy lub też wypowiadania pewnych słów. Przekonanie o wartości i celowości takich zachowań wynika z wiary, że są one święte lub nieczyste w sensie rytualnym. Wskutek złamania tabu następuje stan skażenia związany z odpowiednimi sankcjami, głównie ze strony sił nadprzyrodzonych. Instytucja tabu została odkryta przez J. Cooka w 1771 wśród tubylców wyspy Tonga.

2) przedmiot, osoba, zwierzę, słowo, miejsce lub czynność objęte zakazem.

3) w psychoanalizie zakaz ograniczający realizację dążeń seksualnych, którego konsekwencją jest stłumienie i ukrycie w podświadomości tendencji do zachowań objętych zakazem.

4) w szerokim sensie tabu oznacza jakikolwiek zakaz.

- Początek dały prace Glovera z lat 70.
- TS rozumie się jako nakładanie ograniczeń; tu jednak chodzi raczej o wykorzystanie ograniczeń do kierowania przeszukiwania w obiecujące rejony przestrzeni dopuszczalnych rozwiązań.

Tabu search

- Podstawy działania
  Ruch jest funkcją transformującą dane rozwiązanie w inne rozwiązania. Podzbiór ruchów generuje dla danego rozwiązania podzbiór rozwiązań zwany *sąsiedztwem*. Na każdym stopniu iteracji przeszukiwane jest sąsiedztwo w celu wyboru najlepszego rozwiązania, które staje się bazowym dla następnego kroku. Ostatnio wykonany ruch jest wprowadzany na *listę zabronionych ruchów*, które nie mogą być wykonane w ciągu najbliższych *s* iteracji. Mechanizm ten umożliwia wyjście z lokalnego minimum i chroni przed cyklicznością ruchów. Niekiedy możliwe jest wykonanie zabronionego ruchu, o ile zdefiniowana *funkcja aspiracji* określi zyskowność takiego ruchu. Koniec obliczeń następuje w momencie przekroczenia limitu czasu, po wykonaniu dopuszczalnej liczby iteracji albo po wykonaniu dopuszczalnej liczby ruchów bez poprawy wartości funkcji kryterium.

Tabu search - przykład

- Sytuacja wejściowa

  Długość uszeregowania = 60

2	5	7	3	4	6	1

- Ruch typu wymiana (ang. swap, exchange)

i=5, j=6

2	6	7	3	4	5	1

Długość uszeregowania = 62

Tabu search - przykład

- Lista tabu (ruchów zabronionych)
  Lista wskazuje, jakie ruchy i jak długo są zabronione

$s=3$

	2	3	4	5	6	7
1						
2						
3				3		
4						
5						
6						1

zamiana (3,5) jest zabroniona w ciągu trzech kolejnych iteracji

zamiana (6,7) jest zabroniona przez jeszcze jedną iterację

- Funkcja aspiracji określa warunki akceptacji zabronionego ruchu np. kryterium *najlepsze rozwiązanie*

Tabu search - przykład

Iteracja 0

2	5	7	3	4	6	1

Długość uszeregowania = 60

Lista tabu

	2	3	4	5	6	7
1						
2						
3						
4						
5						
6						

Top 5

Ruch	Z
5,4	-6
7,4	-4
3,6	-2
2,3	0
4,1	-1

Tabu search - przykład

Iteracja 1 - ruch (5,4)

2	4	7	3	5	6	1

Długość uszeregowania = 60-6=54

Lista tabu

	2	3	4	5	6	7
1						
2						
3						
4				3		
5						
6						

Top 5

Ruch	Z
3,1	-2
2,3	-1
3,6	1
7,1	2
6,1	4

Tabu search - przykład

Iteracja 2 - ruch (3,1)

2	4	7	1	5	6	3

Długość uszeregowania = 54-2=52

Lista tabu

	2	3	4	5	6	7
1		3				
2						
3						
4				2		
5						
6						

Top 5

Ruch	Z
1,3	2
2,4	4
7,6	6
4,5	7
5,3	9

Tabu search - przykład

Iteracja 3 - ruch (2,4)

4	2	7	1	5	6	3

Długość uszeregowania = 54+4=58

Lista tabu

	2	3	4	5	6	7
1		2				
2			3			
3						
4				1		
5						
6						

Top 5

Ruch	Z
4,5	-6
5,3	-2
7,1	0
1,3	3
2,6	6

Tabu search - przykład

Iteracja 4 - ruch (4,5)

5	2	7	1	4	6	3

Długość uszeregowania = 58-6=52

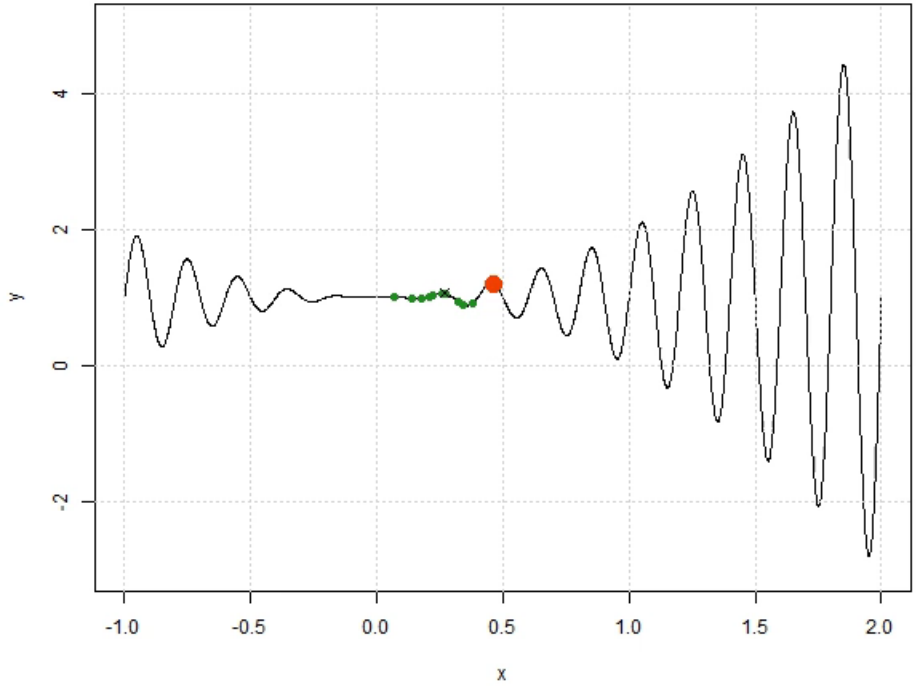
Lista tabu

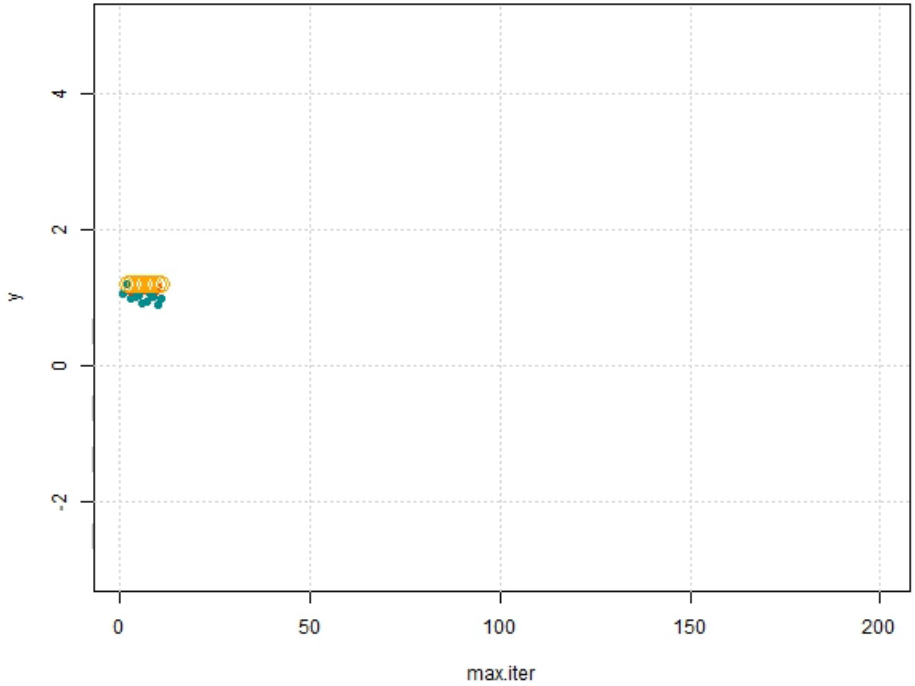
	2	3	4	5	6	7
1		1				
2			2			
3						
4				3		
5						
6						

Top 5

Ruch	Z
7,1	0
4,3	3
6,3	5
5,4	6
2,6	8

Tabu search - przykład





Tabu search

```
BEGIN
  R := R0 {rozwiązanie początkowe};
  TL := empty {lista tabu jest pusta};
  R* := R0; C* := C(R0) {najlepsze rozwiązanie i funkcja celu};
  WHILE (nie spełniony warunek zatrzymania) DO
  BEGIN
        C := MAX;
        WHILE (nie przetestowane sąsiedztwo rozwiązania R) DO
        BEGIN
          IF m(R) NOT tabu THEN BEGIN
            R' := m(R) {nowe rozwiązanie po wykonaniu ruchu m};
            IF C(R') < C THEN   BEGIN  C := C(R'); R'' := R'   END;
          END
        END;
        IF C < C* THEN  BEGIN C* := C; R* := R" END;
        R := R";
        aktualizacja TL;
  END;
END.
```

Metody badania algorytmów

- Dokładność algorytmu określa oddalenie otrzymywanego rozwiązania od optymalnego. Ocena dokładności ma podstawowe znaczenie w przypadku algorytmów heurystycznych, gdyż algorytmy te nie dają rozwiązań o takiej samej dokładności dla problemów o różnych rozmiarach, co więcej – nie dają rozwiązań powtarzalnych. Powszechnie stosowane są trzy metody:
  1. oszacowanie zachowania się w najgorszym przypadku,
  2. ocena probabilistyczna,
  3. ocena eksperymentalna (wykorzystywana najczęściej).

Metody badania algorytmów

- Oszacowanie zachowania się w najgorszym przypadku jest metodą skomplikowaną, a przy tym nie oddającą w pełni najgorszego zachowania się algorytmu dla przeciętnych danych. Dlatego nadaje się jedynie dla stosunkowo prostych algorytmów.
- Analiza probabilistyczna algorytmów przybliżonych dostarcza informacji dotyczących ich średniego zachowania. Jest bardzo złożona z uwagi na konieczność analitycznego wyznaczenia względnego i bezwzględnego błędu algorytmu, stąd do tej pory przeanalizowano tą metodą niewiele algorytmów.
- Niedogodności dwóch pierwszych metod decydują o tym, że najchętniej stosuje się metodę eksperymentalną oceny algorytmów przybliżonych. Ocena eksperymentalna służy do badania przeciętnego zachowania się rozwiązań tworzonych przez algorytmy przybliżone. Polega ona na porównaniu rozwiązań generowanych przez dany algorytm z innymi rozwiązaniami (najlepiej - optymalnymi) dla reprezentatywnej próby rozważanego problemu.