

www.kwmimkm.polisi.pl

METODY HEURYSTYCZNE

wykład 2

www.kwmimkm.polisi.pl

STRATEGIE HEURYSTYCZNE

www.kwmimkm.polisi.pl

METODA WZROSTU (SIMPLE) HILL-CLIMBING

Operator - działanie podejmowane w stosunku do stanu aktualnego aby otrzymać z niego kolejny stan.

- Wygeneruj stan początkowy (CS).
- Jeśli CS jest stanem końcowym to zwróć **SUKCES** i zakończ.
- Wybierz operator, który nie był jeszcze używany dla stanu CS i wygeneruj przy jego pomocy stan NS.
- Jeżeli:
 - NS jest stanem końcowym to zwróć **SUKCES** i zakończ;
 - NS nie jest stanem końcowym, ale jest lepszy od CS to CS=NS.
- Powróć do 3.

Strategia stosowana do zagadnień ciągłych oraz dyskretnych.

www.kwmimkm.polisi.pl

HC: PROBLEM 8 KRÓLOWYCH

Lokalne minimum:

- Stan ma 1 konflikt.
- Każde przesunięcie w kolumnie zwiększa liczbę konfliktów.

www.kwmimkm.polisi.pl

METODA NAJSZYBSZEGO WZROSTU (STEEPEST-ASCENT HILL-CLIMBING)

- Wygeneruj stan początkowy (CS).
- Jeśli CS jest stanem końcowym to zwróć **SUKCES** i zakończ.
- Dla każdego operatora, który nie był jeszcze używany wygeneruj stan NS.
- Jeżeli:
 - NS jest stanem końcowym: zwróć **SUKCES** i zakończ.
 - NS nie jest stanem końcowym, ale jest lepszy od CS to CS=NS.
- Powróć do 3.

- W SHC przechodzi się ze stanu bieżącego do pierwszego stanu dla którego wartość funkcji oceniającej okazała się lepsza.
- W SAHC sprawdza się dla każdego stanu wszystkie operatory i wybiera się ten, który daje najlepszy nowy stan.

www.kwmimkm.polisi.pl

WADY METODY HILL CLIMBING

- Rozwija jedną ścieżkę.
- W przypadku jednakowych kosztów losuje ścieżkę.
- Lokalne ekstrema.
- Plateau (równiny).
- Wąskie grzbiety.

Co zrobić?

- Powrót do poprzednich węzłów i próba przemieszczania się w innym kierunku lub metoda wielostartu.
- Wykonanie dużego skoku w pewnym kierunku (eksplorowanie nowego fragmentu przestrzeni stanów).
- Wykonanie ruchu w kilku kierunkach naraz.

PIERWSZY NAJLEPSZY (BEST FIRST)

www.kwmkm.pol.pl

- W wersji podstawowej tożsamy z HC dla wariantu dyskretnego.
- W procesie poszukiwania zawsze można wrócić do węzła lepszego (jeśli są pamiętane).

Wersje:

- szukanie zachłanne;
- A*.



7

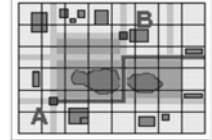
SZUKANIE ZACHŁANNE (GREEDY SEARCH)

www.kwmkm.pol.pl

- Jedna z najprostszych strategii BF.
- F. heurystyczna szacuje pozostałe koszty dotarcia do celu (jak: odległość od celu).
- Najpierw rozwijany jest najbliższy węzeł.

Problemy szukania drogi: metryki:

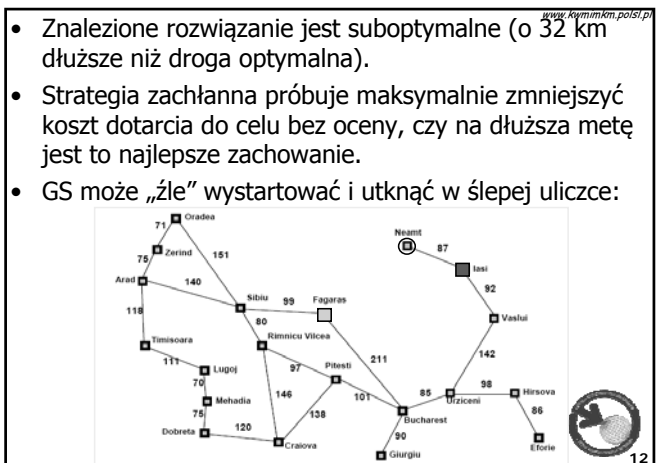
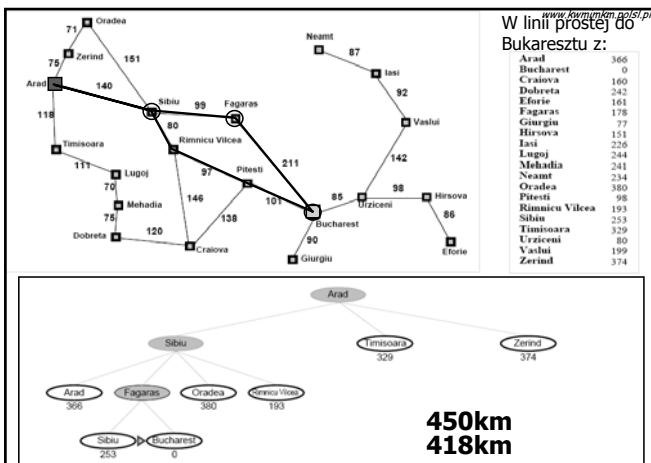
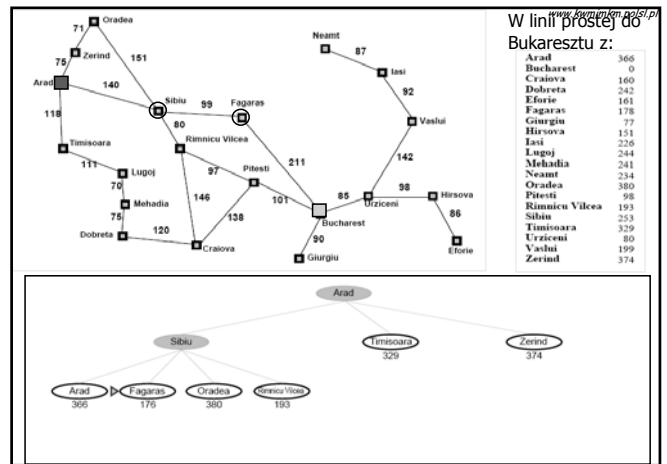
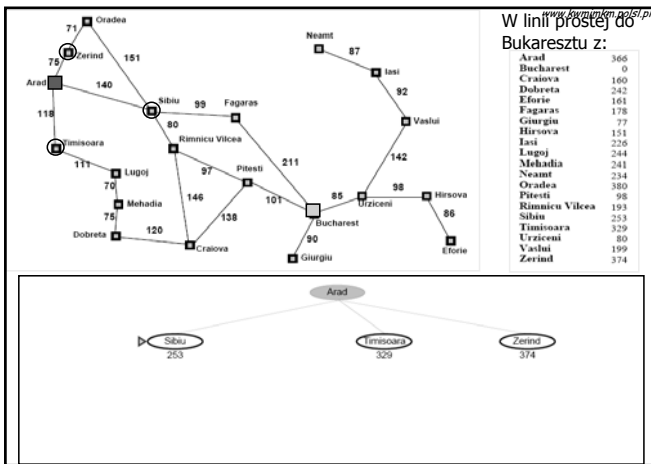
- Euklidesowa;
- Manhattan (poruszanie się tylko po liniach poziomych i pionowych).



W innych problemach: oceny podobieństwa.



8



- Znalezione rozwiązanie jest suboptymalne (o 32 km dłuższe niż droga optymalna).
- Strategia zachłanna próbuje maksymalnie zmniejszyć koszt dotarcia do celu bez oceny, czy na dłuższą metę jest to najlepsze zachowanie.
- GS może „źle” wystartować i utknąć w ślepej uliczce:

12

- GS nie jest algorytmem optymalnym (niekoniecznie znajduje rozwiązanie o minimalnym koszcie).
- GS nie jest algorytmem zupełnym (nie zawsze znajduje rozwiązanie).
- Złożoność czasowa GS w najgorszym razie wynosi $O(b^m)$, dla m kroków w głąb i średnio b możliwości.
- Złożoność pamięciowa GS wynosi $O(b^m)$ (przechowuje wszystkie węzły w pamięci).
- Alternatywa: minimalizacja kosztów dojścia do danego węzła $g(n)$ – metoda zupełna i optymalna, lecz mało efektywna...
- Metoda A* łączy obydwa podejścia.



STRATEGIA A*

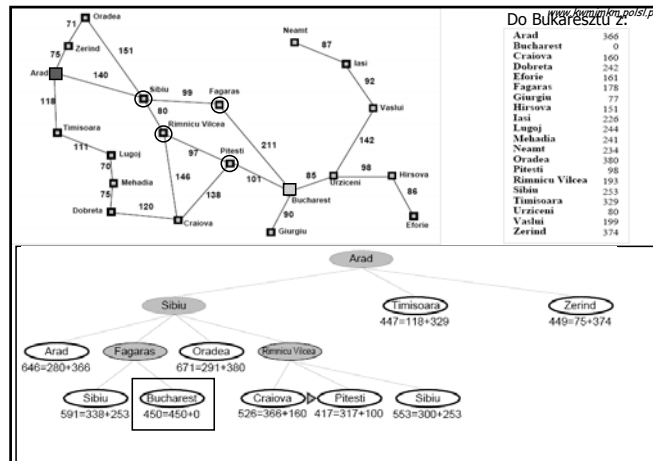
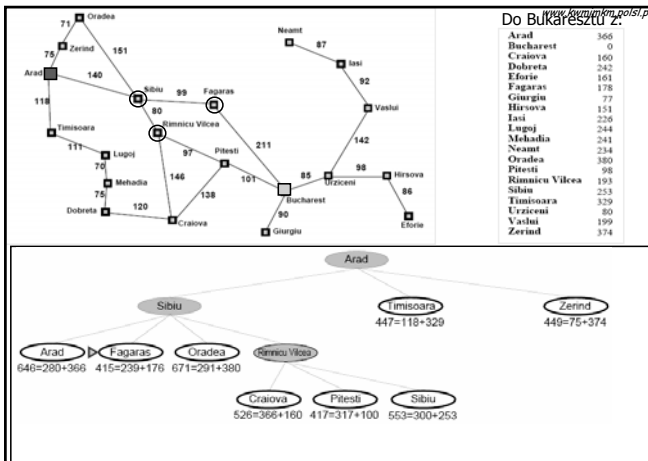
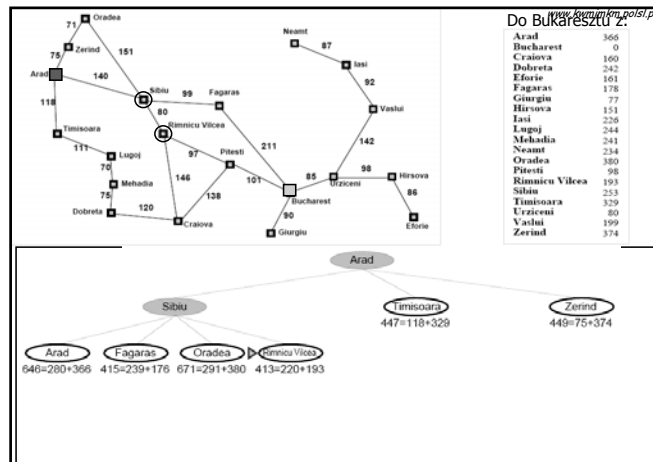
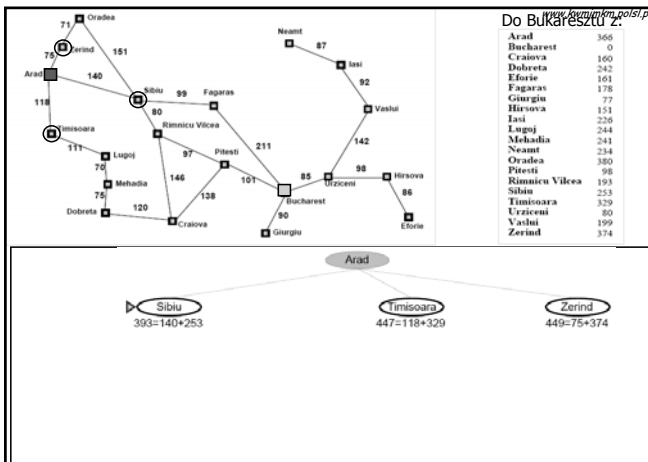
Funkcji heurystyczna oceniająca koszty najtańszego rozwiązania przechodzącego przez węzeł n :

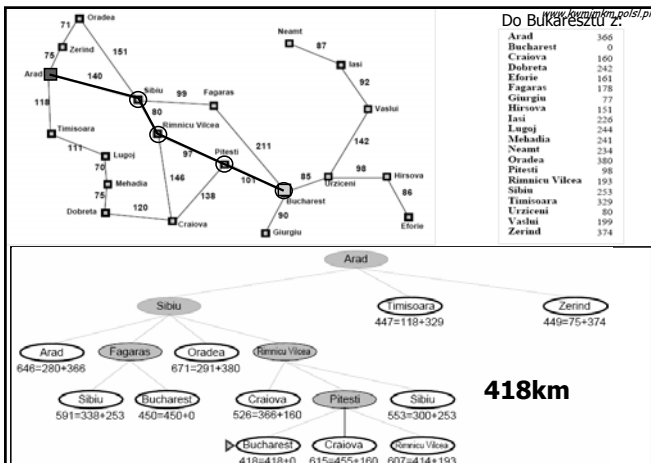
$$f(n) = g(n) + h(n)$$

$g(n)$ – dotychczasowy koszt dotarcia do stanu n ;

$h(n)$ – oszacowanie kosztu od stanu bieżącego n do stanu docelowego.

1. Rozpocznij od węzła początkowego i twórz nowe węzły $\{n\}$ dopóki cel nie zostanie osiągnięty;
2. Posortuj nowe węzły $\{n\}$ zgodnie z funkcją: $f(n)=g(n)+h(n)$.
3. Odrzuć ścieżki zapłtane.
4. Wybierz najlepszy węzeł n'
5. Zostaw tylko najtańszą ścieżkę do n' .
6. Jeśli n' jest celem skończ; jeśli nie, rozwijaj dalsze węzły $\{n\}$ łącznie z n'





- A* jest algorytmem optymalnym (znajduje rozwiązanie o minimalnym koszcie) jeśli heurystyka h jest dopuszczalna.
- A* jest algorytmem zupełnym (zawsze znajduje rozwiązanie) jeśli nie ma nieskończenie wielu stanów takich, że: $f \leq f(G)$ gdzie: G - stan docelowy.
- Złożoność czasowa: wykładnicza względem [błąd względny $h \times$ dług. rozwiązania].
- Złożoność pamięciowa: $O(b^m)$ (przechowuje wszystkie węzły w pamięci)

DOPUSZCZALNOŚĆ HEURYSTYKI

Funkcja heurystyczna $h(n)$ jest dopuszczalna, jeśli w każdym stanie n spełnia następujący warunek:

$$h(n) \leq h^*(n)$$

$h^*(n)$ - rzeczywisty koszt ścieżki od stanu n do celu.

Np.:

- A – stan początkowy;
- F – stan docelowy;
- Na krawędziach: rzecz. koszty przejścia między stanami;
- Wartości f . heurystycznej $h(n)$:

	A	B	C	D	E	F
$h(-)$	10	3	7	4	2	0

Np.:

$h_{SLD}(n)$:

Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

F. odległości w linii prostej $h_{SLD}(n)$ jest dopuszczalna: - nigdy nie przekracza rzeczywistej odległości drogowej.

SPÓJNOŚĆ HEURYSTYKI

Dopuszczalna funkcja heurystyczna $h(n)$ jest spójna (spełnia wymóg monotoniczności), jeżeli dla każdego węzła n i każdego jego następcy n' :

$$h(n) \leq c(n, n') + h(n')$$

Jeżeli heurystyka $h(n)$ jest spójna, to A* znajdzie optymalną ścieżkę.

- ### STRATEGIA IDA* (ITERATIVE DEEPENING A*)
1. Stosuj algorytm szukania w głąb.
 2. Oceniaj całkowite koszty $f(n) = g(n) + h(n)$ heurystyką A*.
 3. Jeśli $f(n) > T$ cofaj się.
 4. Jeśli nie znaleziono rozwiązania zwiększ T i powtarzaj.
- Obcinane są węzły, których wartość funkcji oceny leży poniżej progu T .
 - Poziom obcinania jest iteracyjnie zwiększany, aż do osiągnięcia celu.
 - Zalety: niewielkie wymagania pamięciowe - jak w szukaniu w głąb - jednakże szybsza.

SYMULOWANE WYŻARZANIE

www.kwmimkm.polisi.pl

(SIMULATED ANNEALING)

- Algorytm z rodziny algorytmów „Generuj i Testuj”.
- Odmiana algorytmu największego wzrostu, w której rozwiązuje się problem lokalnych ekstremów.
- Dopuszcza się przejścia ze stanu bieżącego także do stanów gorszych
(uniezależnienie się od punktu startowego i badanie znacznie większego obszaru przestrzeni rozwiązań).
- W miarę upływu czasu prawdopodobieństwo wykonania „złego” posunięcia zmniejsza się.



25

Z metalurgii:

www.kwmimkm.polisi.pl

- Kawałek metalu ogrzewany i pozostawiany do powolnego ostygnięcia.
- Powolne i regularne chłodzenie - obniżenie poziomu energii atomów do momentu znalezienia się w stanie metastabilnym (o minimalnej energii).
- Gwałtowne ochłodzenie: zamrożenie atomów na przypadkowych pozycjach.



26

Prawdopodobieństwo przejścia do wyższego stanu energetycznego:

www.kwmimkm.polisi.pl

$$p = e^{-\Delta E / kT}$$

ΔE – zmiana energii stanu;

T – temperatura przy jakiej następuje przejście do wyższego stanu energetycznego;

k – stała Boltzmanna.



27

SZTUCZNE SYMULOWANE WYŻARZANIE

www.kwmimkm.polisi.pl

$$p' = e^{-\Delta E / T}$$

ΔE – zmiana funkcji heurystycznej;

T – temperatura (tempo studzenia, rozkład wyżarzania).

- Rozkład wyżarzania musi być dany (założony).
- Najczęściej jest związany z czasem (krokiem) procesu.
- Prawdopodobieństwo małego skoku w kierunku wzrastającej energii jest większe niż dużego skoku.
- Prawdopodobieństwo skoku powodującego wzrost energii zmniejsza się wraz ze spadkiem temperatury.
- Wielkość skoków maleje z upływem czasu.



28

1. Wygeneruj stan początkowy (CS).

2. Ustaw początkową temperaturę T oraz stan BestSoFar:=CS.

3. Dopóki $T > 0$ i są stany nie wykorzystane jako CS:

- Wybierz jeden z niewykorzystanych stanów i nazwij go NS.
- Oblicz $\Delta E = E(NS) - E(CS)$.
- Jeśli $\Delta E \leq 0$ to CS:=NS oraz BestSoFar:=NS.
- Jeśli $\Delta E > 0$ to CS:=NS z prawdopodobieństwem p .
- Zmniejsz temperaturę T .

4. Zwróć BestSoFar jako rozwiązanie i zakończ.

www.kwmimkm.polisi.pl



29

GRY DWUOSOBOWE

www.kwmimkm.polisi.pl

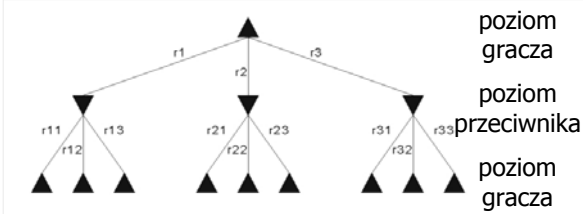
Typowe założenia:

1. W grze uczestniczy dwóch graczy.
2. Gracze wykonują ruchy naprzemiennie.
3. W każdej sytuacji na planszy jest skończona liczba możliwych do wykonania ruchów.
4. Sytuacja na planszy i wykonany ruch jednoznacznie wyznaczają następną sytuację na planszy.
5. Każda możliwa sytuacja na planszy może być jednoznacznie zaklasyfikowana do jednej z następujących kategorii:
 - wygrana pierwszego gracza,
 - wygrana drugiego gracza,
 - remis,
 - sytuacja nierozstrzygnięta.



30

- Pojedyncza partia gry może być w pełni opisana przez ciąg naprzemiennych ruchów obu graczy, od początkowego ustawienia do rozstrzygnięcia.
- Aby w dowolnym momencie partii wybrać najbardziej odpowiedni ruch dla jednego z graczy, można rozważyć wszystkie możliwe scenariusze jej dalszego ciągu.
- Naturalna reprezentacja: drzewo gry.

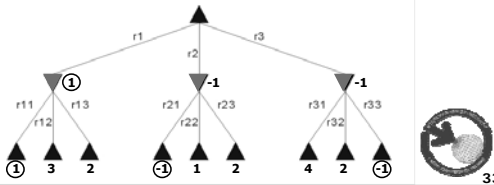


- Obecność przeciwnika, którego ruchy nie mogą być przewidziane i dokładnie zaplanowane, decyduje o istotnej odmienności zadania przeszukiwania w grach od innych zadań przeszukiwania.
- Istnieje zatem konieczność posługiwania się specyficznymi algorytmami.
- Najczęściej stosowane podejście: przyjęcie założenia, że przeciwnik dążąc do wygrania partii zawsze wybiera najkorzystniejszy dla siebie (a więc najmniej korzystny dla gracza) ruch.



Algorytm mini-max

- Najprostszy algorytm przeszukiwania drzew mający zastosowanie w nielosowych algorytmach gry w dwie osoby, np. szachów.
- Dla każdego ruchu symulujemy wszystkie możliwe ruchy przeciwnika.
- Za wartość danego naszego ruchu uznajemy wartość najlepszego z punktu przeciwnika ruchu, który może on wykonać, jeśli my wykonamy dany ruch (część min).
- Następnie wybieramy ruch o najwyższej wartości (część max).



Pełny mini-max

1. Przypisz liściom drzewa ocenę jako użyteczność z punktu widzenia aktualnego gracza;
2. Dla każdego poziomu k , drzewa, zaczynając od poziomu przedostatniego i kończąc na poziomie 0,
 1. jeśli poziom k odpowiada aktualnemu graczowi, przypisz każdemu węzłowi tego poziomu ocenę wyznaczoną jako maksimum ocen jego węzłów potomnych z poziomu $k+1$;
 2. jeśli poziom k odpowiada przeciwnikowi aktualnego gracza, przypisz każdemu węzłowi tego poziomu ocenę wyznaczoną jako minimum ocen jego węzłów potomnych z poziomu $k+1$;
3. Wybierz ruch prowadzący do węzła poziomu 0 o maksymalnej ocenie.

Tak sformułowany algorytm zakłada, że dysponujemy zbudowanym pełnym drzewem gry – zwykle nie jest to możliwe...



Obcięty mini-max

- Ograniczenie głębokości analizowania drzewa gry do pewnej liczby poziomów (zależnej od złożoności gry i dostępnej mocy obliczeniowej).
- Wystąpią węzły terminalne odpowiadające nierozstrzygniętej partii – należy je w jakiś sposób ocenić.
- Ich ocena nie może być oparta na wyniku partii (brak rozstrzygnięcia) ani na ocenach węzłów potomnych (brak węzłów potomnych), lecz wyłącznie na analizie związanego z nim stanu gry – potrzebna jest funkcja heurystyczna...
- F. heurystyczna ocena użyteczność danego stanu z punktu widzenia gracza bez rozważania dalszych możliwych ruchów.



Algorytm mini-max – zadowalający, jeśli stosowany do umiarkowanie złożonych gier albo wymagamy tylko umiarkowanego poziomu gry (porównywalnego ze średnim ludzkim graczem).

[http://wazniak.mimuw.edu.pl/index.php?title=Sztuczna_inteligencja_w_module_„Gry_dwuosobowe”_–_postaci_algorytmow_mini-max_i_obciety_mini-max_\(wersja_rekurencyjna\).](http://wazniak.mimuw.edu.pl/index.php?title=Sztuczna_inteligencja_w_module_„Gry_dwuosobowe”_–_postaci_algorytmow_mini-max_i_obciety_mini-max_(wersja_rekurencyjna).)



Cięcia alfa-beta

www.kwmimkm.polisi.pl

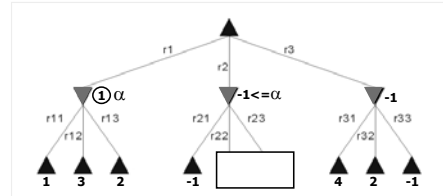
- Oparty na obciętych algorytmie mini-maks, uwzględniającym kryteria umożliwiające bezpieczne pominięcie (wycięcie) fragmentów drzewa gry.
- Rezygnacja z rozbudowy i analizy drzewa gry tam, gdzie jej wynik i tak nie wpłynie na wybierany ruch.
- Oszczędność zasobów, które można wykorzystać do przeszukiwania obiecujących fragmentów drzewa gry.



37

Cięcie alfa: oceniając węzeł przez maksymalizację ocen węzłów potomnych możemy zakończyć wyznaczanie oceny węzła potomnego natychmiast po stwierdzeniu, że musi być ona niższa niż dotychczasowe maksimum α .

Cięcie beta: oceniając węzeł przez minimalizację ocen węzłów potomnych możemy zakończyć wyznaczanie oceny węzła potomnego natychmiast po stwierdzeniu, że musi być ona wyższa niż dotychczasowe minimum β .



38

- Skuteczność algorytmów opartych na zasadzie mini-maksu w dużej mierze zależy od jakości używanej funkcji heurystycznej.

- Funkcja heurystyczna powinna:

1. Przypisywać ocenę dodatnią stanowi, z którego większe szanse wygranej ma gracz,
2. Przypisywać ocenę ujemną stanowi, z którego większe szanse wygranej ma przeciwnik,
3. Przypisywać ocenę tym większą co do wartości bezwzględnej, im przewaga szans wygranej gracza albo przeciwnika jest większa.



39

- W najskuteczniejszych programach grających w gry (zwłaszcza szachy) - niezwykle wyrafinowane funkcje heurystyczne, oparte na najwyższej klasy wiedzy eksperckiej i wielu eksperymentach.

- **Dobre podejście:** by funkcja heurystyczna (nawet bardzo skuteczna) mogła podlegać modyfikacjom na podstawie rozgrywanych partii (wyposażanie programów grających w gry w zdolność do uczenia się).



40

DOBÓR METODY HEURYSTYCZNEJ

www.kwmimkm.polisi.pl



41

Pytania:

1. Czy można dokonać dekompozycji problemu?
2. Czy można ignorować/anulować pewne stany?
3. Czy wynik jest znany z góry?
4. Czy rozwiązanie jest względne czy bezwzględne?
5. Czy oczekiwane rozwiązanie jest stanem czy ścieżką?
6. Jaka ilość wiedzy jest wymagana?
7. Czy program wymaga interakcji użytkownika?



42

Ad. 1. Dekompozycja problemu.

www.kwmimkn.polsl.pl

Podział zadania na mniejsze, łatwiejsze do rozwiązania fragmenty.

Korzyści:

- Możliwość rozwiązania dużych zadań, których nie można rozwiązać bez ich podziału;
- Mniejsze problemy można zazwyczaj rozwiązać w krótszym czasie. Sumaryczny czas obliczeń po dekompozycji jest często krótszy niż dla zadania oryginalnego.

Nie wszystkie problemy dają się dekomponować...



43

Ad. 2. Ignorowanie i anulowanie kroków.

www.kwmimkn.polsl.pl

Klasy zadań:

1. Z ignorowaniem błędów - np. dowodzenie twierdzeń: błędnie wykonane wnioskowanie można zignorować i poszukiwać innej drogi rozwiązania).
2. Odwracalne - np. ósemka: możliwy jest powrót z niewłaściwego stanu do poprzedniego, co daje szansę na znalezienie rozwiązania).
3. Nieodwracalne - np. szachy. Algorytm musi podejmować decyzje obciążone ryzykiem. Niezbędna analiza w przód (planowanie)



44

Ad. 3. Znajomość wyniku.

www.kwmimkn.polsl.pl

1. **Pewny wynik** (np. ósemka)
2. **Niepewny wynik** (np. szachy, brydż)

Ad. 4. Względność rozwiązania.

1. Rozwiązanie względne:

problemy typu „dowolna ścieżka do rozwiązania” – rozwiązywalne w rozsądnym czasie przy zastosowaniu dobrej heurystyki.

2. Rozwiązanie bezwzględne:

Problemy typu „najlepsza ścieżka do rozwiązania” wymagające bardziej wyczerpującego przeszukiwania przestrzeni stanów.



45

Ad. 5. Typy rozwiązań.

www.kwmimkn.polsl.pl

1. **Pojedynczy stan z przestrzeni** (istotny jest stan końcowy a nie wyniki pośrednie).
2. **Ścieżka prowadząca od stanu początkowego do rozwiązania** (wynik jest znany, istotna jest droga dojścia do niego).

Ad. 6. Wiedza.

1. **Problemy wymagające wiedzy do mechanizmu sterowania** (wiedza niezbędna by usprawnić algorytm sterowania i eliminować niepożądane stany). Np szachy: ruchy figur, strategia.
2. **Problemy wymagające szerokiej wiedzy** (wiedza ma pomóc w określeniu rozwiązania, nie tylko przyspieszeniu jego znalezienia). Np. rozumienie tekstu.



46

Ad. 7. Interakcja z użytkownikiem.

www.kwmimkn.polsl.pl

1. Systemy samodzielne:

Komputer szuka rozwiązania bez komunikacji z użytkownikiem i nie wymaga się wyjaśniania procesu dochodzenia do rozwiązania).

Np.: dowodzenie twierdzeń.

2. Systemy konwersacyjne:

Wymagana jest interakcja z użytkownikiem.

Np.: systemy ekspertowe i diagnozy medyczne.



47

OPTIMALIZACJA

www.kwmimkn.polsl.pl

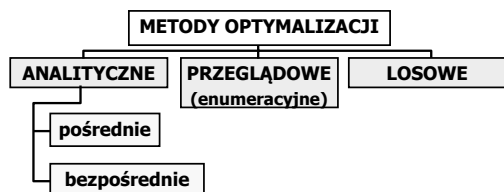


48

OPTIMALIZACJA:

„Działanie, mające na celu zwiększenie efektywności aż do osiągnięcia pewnego optimum”.

- Cel główny: ulepszenie.
- Cel drugorzędny: osiągnięcie optimum.



49

Metody analityczne bezpośrednie:

- Poruszanie się po wykresie funkcji w kierunku wyznaczonym przez lokalny gradient (wspinaczka po najbardziej stromym zboczu z możliwych).

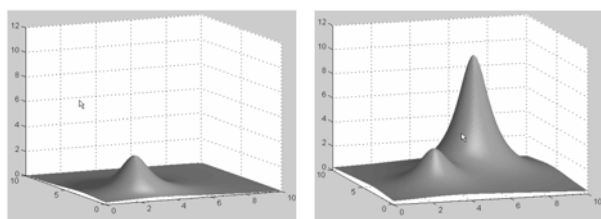
Metody analityczne pośrednie:

- Poszukiwanie ekstremów lokalnych poprzez rozwiązanie układu równań (zwykle nieliniowych), otrzymanych poprzez przyrównanie gradientu funkcji celu do zera.



50

GŁÓWNA WADA METOD ANALITYCZNYCH: MAŁA ODPORNOŚĆ

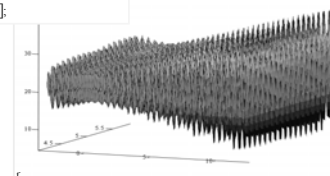


51

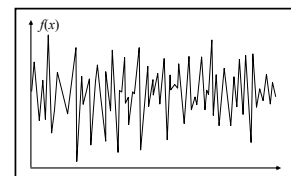
Funkcja trudna do optymalizacji metodami analitycznymi:

$$f(x_1, x_2) = 21.5 \cdot \sin(4\pi x_1) + x_2 \cdot \sin(20\pi x_2)$$

$$x_1 \in [-3.0, 12.1]; x_2 \in [4.1, 5.8]$$



Funkcja niemożliwa do optymalizacji metodami analitycznymi:



Czasem maksimum globalne nie jest pożądane:



Preferowane są czasem rozwiązania, których otoczenie przyjmuje wartości bliskie temu ekstremum a nie te, dla których niewielkie oddalenie się od ekstremum powoduje gwałtowny spadek wartości funkcji.

Np: w przypadku inwestycji kapitałowych, by nie ryzykować straty z powodu niezbyt precyzyjnie zdefiniowanej funkcji, bądź nieznaczącej zmiany jakiegoś parametru funkcji.



53

METODY ENUMERACYJNE:

- Sprowadzają się do przeszukiwania wszystkich punktów przestrzeni w poszukiwaniu optimum.
- Algorytm niezwykle prosty lecz skuteczny jedynie w przypadku skończonych, małych przestrzeni.
- Zwykle sprawdzenie wszystkich możliwości jest niemożliwe w rozsądnym czasie (tzw. przekleństwo wymiaru).



54

METODY LOSOWE:

- W swej najprostszej postaci: bada się losowo całą przestrzeń zadania nie korzystając z innych informacji.
- Poszukiwanie takie jest zwykle bardzo czasochłonne (zwykle jednak mniej niż metody enumeracyjne).

Algorytmy genetyczne i ewolucyjne również zawierają element losowości (algorytm zrandomizowany).

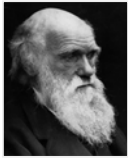


55

ALGORYTMY GENETYCZNE



56

Karol Darwin (1859 „On the origin of species”):

Gregor Johann **Mendel** (1822-1884) - austriacki zakonnik, augustianin, prekursor genetyki.

- Sformułował podstawowe prawa dziedziczenia (3 prawa Mendla), przeprowadzając badania nad krzyżowaniem roślin, głównie grochu jadalnego.



- Ewolucja przez dobór naturalny oznacza, że przeżywają i rozmnażają się osobniki najlepiej przystosowane do warunków środowiska.
- Wynik przystosowania zależy od:
 - organizmu;
 - środowiska.



- Na świat przychodzi dużo więcej potomstwa, niż może pomieścić środowisko.
- Przeżywają nieliczni, ale za to najlepsi (selekcja naturalna).
- W procesie ewolucji istotne jest zachowywanie różnorodności cech.
- Siła ewolucji to nie zaawansowany proces doskonalenia jednostki, lecz utrzymywanie dużej liczby różnorodnych osobników (tzw. populacji), która ewoluuje jako całość.



59

ALGORYTMY GENETYCZNE:

- AG odwzorowują naturalne procesy ewolucji zachodzące w czasie.
- Ich celem jest maksymalne dopasowanie osobników do istniejących warunków życia.
- Rolę środowiska spełnia tu funkcja oceniająca (funkcja celu).
- Pomimo elementu losowości AG nie błądzą przypadkowo, lecz wykorzystują efektywnie przeszłe doświadczenia.



60

John H. Holland, 1975:

„Adaptation in Natural and Artificial Systems“:

- Koncepcja algorytmu przeszukiwania opartego na zasadzie doboru naturalnego.
- Procedurę probabilistycznego przeszukiwania dyskretnej przestrzeni stanów nazwał **algorytmem genetycznym**.



61

TERMINOLOGIA:

100011011

- gen – najmniejsza składowa chromosomu (decyduje o dziedziczności jednej lub kilku cech);
- chromosom – uporządkowany ciąg genów (ciąg kodowy). Zwykle utożsamiany z osobnikiem;
- locus – miejsce genu w chromosomie;
- allele – warianty (stany) jednego genu warunkujące daną cechę;
- populacja – pewna liczba osobników (chromosomów);



62

TERMINOLOGIA:

100011011

- genotyp – ogół genów danego osobnika;
- fenotyp – ogół cech ujawniających się na zewnątrz (np. rozkodowana postać zmiennych projektowych);
- mutacja – zmiana jednego lub kilku genów w chromosomie;
- krzyżowanie – operacja mająca na celu wymianę materiału genetycznego między osobnikami;
- selekcja – wybór osobników, które zostaną poddane operacjom genetycznym.



63

SCHEMAT DZIAŁANIA AG:

```

procedure algorytm_genetyczny
begin
  t:=0
  wybierz populację początkową P(t)
  oceń P(t)
  while (not warunek_zakończenia) do
    begin
      t:=t+1
      wybierz P(t) z P(t-1) (selekcja)
      zmień P(t) (działanie operatorów genetycznych)
      oceń P(t)
    end
  end

```



64

ZASTOSOWANIA PRAKTYCZNE AG (i AE):

- wyznaczanie trasy połączeń kablowych;
- harmonogramowanie;
- sterowanie adaptacyjne;
- rozgrywanie gier;
- zadanie plecakowe;
- zadanie komiwojażera;
- sterowanie optymalne;
- optymalizacja obsługi pytań w bazach danych;



65

- nieliniowe systemy dynamiczne – analiza danych;
- przewidywanie;
- projektowanie sieci neuronowych: architektury i wagi;
- poruszanie robotem;
- tworzenie programów;
- planowanie;
- znajdowanie kształtu molekuł białek;
- tworzenie grafik i muzyki;
- ...



66