Sztuczna inteligencja. Uzupełnienie do Wykładów 5,6,7. Więzy i gry

Paweł Rychlikowski

Instytut Informatyki UWr

4 maja 2022

Problemy spełnialności więzów. Przypomnienie definicji

Definicja

Problem spełnialności więzów ma 3 komponenty:

1. Zbiór zmiennych $X_1, \ldots, X_n$
2. Zbiór dziedzin (przypisanych zmiennym)
3. Zbiór więzów, opisujących dozwolone kombinacje wartości jakie mogą przyjmować zmienne.

Przykład

Zmienne: X, Y, Z
Dziedziny: $X \in \{1,2,3,4\}, Y \in \{1,2\}, Z \in \{4,5,6,7\}$
Więzy: $X + Y \geq Z, X \neq Y$

Podstawowe narzędzia

1. Propagacja więzów (wnioskowanie ograniczające dziedziny, lub wprowadzające nowe, prostsze więzy)
2. Backtracking (zupełne przeszukiwanie przestrzeni stanów)

Podstawowe narzędzia

1. Propagacja więzów (wnioskowanie ograniczające dziedziny, lub wprowadzające nowe, prostsze więzy)
2. Backtracking (zupełne przeszukiwanie przestrzeni stanów)

Użycie zewnętrznych **solverów**, przykładowo **SWI-Prologa**

Więzy jako graf

Uwaga

Więzy binarne (x R y) są ważną klasą więzów (bo wszystkie więzy można wyrazić jako binarne)

Więzy jako graf

Uwaga

Więzy binarne (x R y) są ważną klasą więzów (bo wszystkie więzy można wyrazić jako binarne)

Uwaga

Więzy binarne możemy przedstawiać jako graf (czasem nazywany siecią więzów)

Więzy jako graf

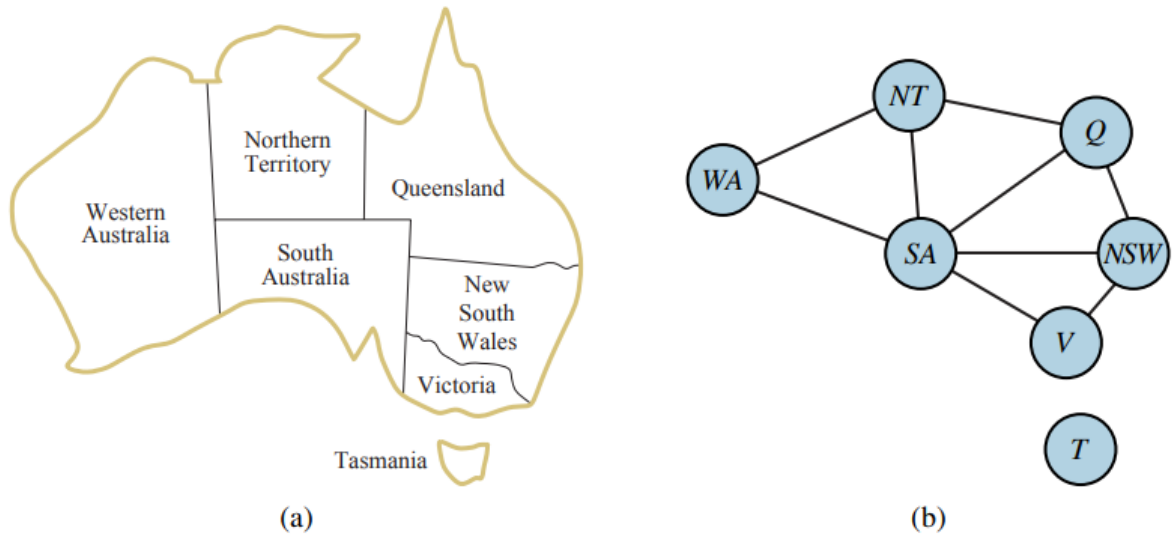
Uwaga

Więzy binarne (x R y) są ważną klasą więzów (bo wszystkie więzy można wyrazić jako binarne)

Uwaga

Więzy binarne możemy przedstawiać jako graf (czasem nazywany siecią więzów)

Przykład dla Australii:



Drzewiasta sieć więzów

Sieć więzów może być drzewem. Wówczas można ją łatwo rozwiązać

Drzewiasta sieć więzów

Sieć więzów może być drzewem. Wówczas można ją łatwo rozwiązać

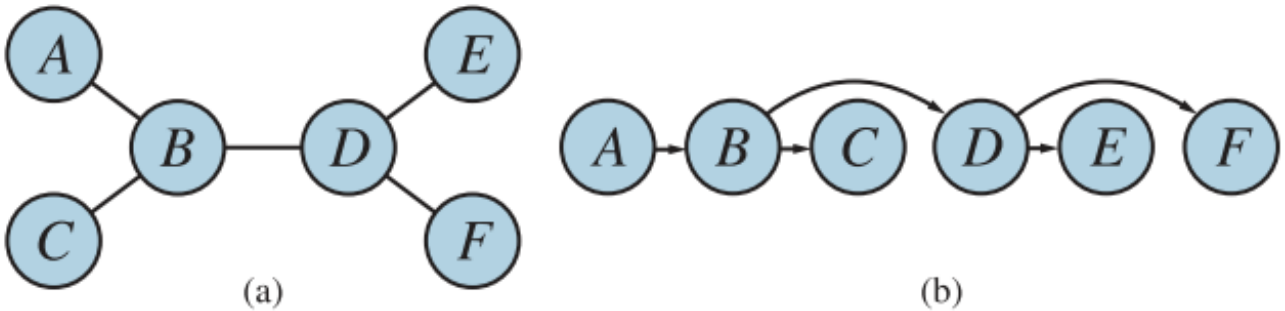


Figure 6.10 (a) The constraint graph of a tree-structured CSP. (b) A linear ordering of the variables consistent with the tree with A as the root. This is known as a **topological sort** of the variables.

Algorytm dla postaci drzewiastej

Rozwiązanie: Algorytm AC-3 wykonywany w porządku topologicznym

Algorytm dla postaci drzewiastej

Rozwiązanie: Algorytm AC-3 wykonywany w porządku topologicznym

function TREE-CSP-SOLVER(*csp*) **returns** a solution, or *failure*
  inputs: *csp*, a CSP with components X, D, C

  $n \leftarrow$ number of variables in X
  assignment $\leftarrow$ an empty assignment
  root $\leftarrow$ any variable in X
  $X \leftarrow$ TOPOLOGICALSORT(X, *root*)
  for $j = n$ **down to** 2 **do**
    MAKE-ARC-CONSISTENT(PARENT(X_j), X_j)
    if it cannot be made consistent **then return** *failure*
  for $i = 1$ **to** n **do**
    assignment$[X_i] \leftarrow$ any consistent value from D_i
    if there is no consistent value **then return** *failure*
  return *assignment*

Figure 6.11 The TREE-CSP-SOLVER algorithm for solving tree-structured CSPs. If the CSP has a solution, we will find it in linear time; if not, we will detect a contradiction.

Jak otrzymać drzewo

- Usuwanie krawędzi (wówczas mamy słabsze więzy)
- **Usuwanie węzłów**

Jak otrzymać drzewo

- Usuwanie krawędzi (wówczas mamy słabsze więzy)
- **Usuwanie węzłów**

Inteligentne usuwanie węzłów: **zgadnij wartość** dla zmiennej

Jak otrzymać drzewo

- Usuwanie krawędzi (wówczas mamy słabsze więzy)
- **Usuwanie węzłów**

Inteligentne usuwanie węzłów: **zgadnij wartość** dla zmiennej
Realistyczne usuwanie węzłów: powtórz powyższe dla każdej wartości dziedziny

Jak otrzymać drzewo

- Usuwanie krawędzi (wówczas mamy słabsze więzy)
- **Usuwanie węzłów**

Inteligentne usuwanie węzłów: **zgadnij wartość** dla zmiennej
Realistyczne usuwanie węzłów: powtórz powyższe dla każdej wartości dziedziny
Dla zbioru (niezbyt licznego) zmiennych: **sprawdź wszystkie kombinacje, spróbuj rozwiązać drzewiaste więzy (Algorytm Cutset-Solver)**

Cutset

Klasyczne zadanie: znajdź możliwie mały zbiór węzłów w grafie, że jego usunięcie zmieni graf w las (**cycle cutset**)

Cutset

Klasyczne zadanie: znajdź możliwie mały zbiór węzłów w grafie, że jego usunięcie zmieni graf w las (**cycle cutset**)

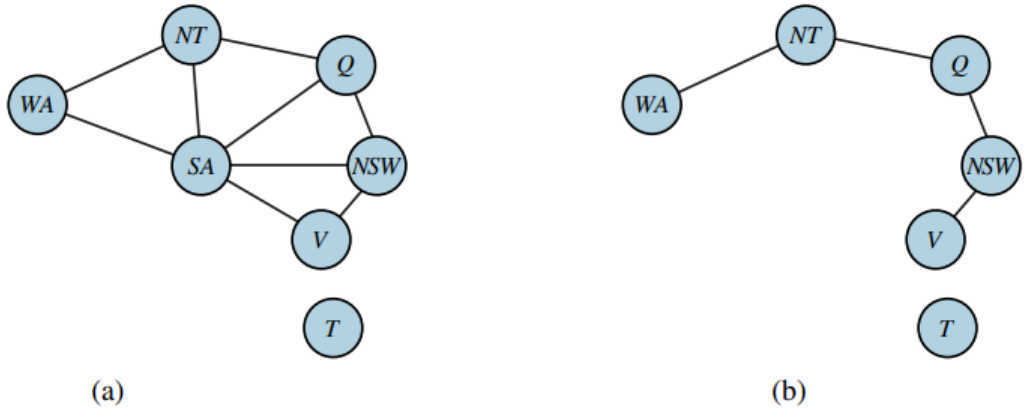


Figure 6.12 (a) The original constraint graph from Figure **??**. (b) After the removal of SA, the constraint graph becomes a forest of two trees.

Cutset

Klasyczne zadanie: znajdź możliwie mały zbiór węzłów w grafie, że jego usunięcie zmieni graf w las (**cycle cutset**)

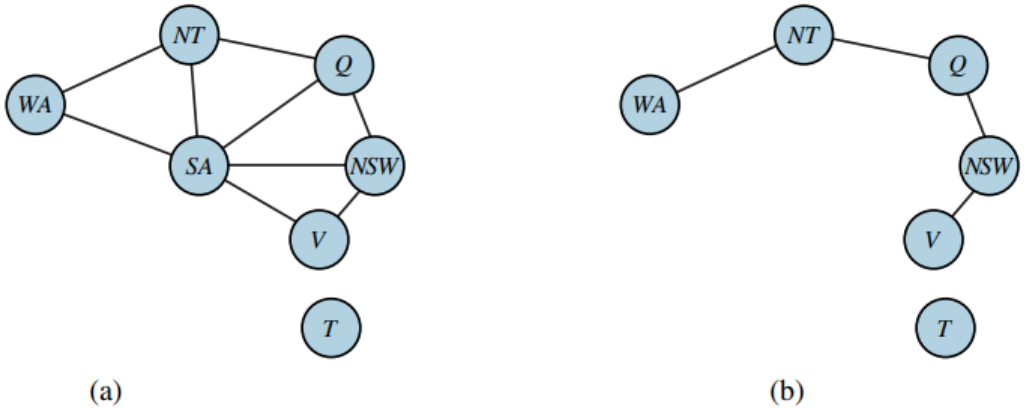


Figure 6.12 (a) The original constraint graph from Figure **??**. (b) After the removal of SA, the constraint graph becomes a forest of two trees.

Dwie wiadomości

- **Zła:** problem jest NP-trudny

Cutset

Klasyczne zadanie: znajdź możliwie mały zbiór węzłów w grafie, że jego usunięcie zmieni graf w las (**cycle cutset**)

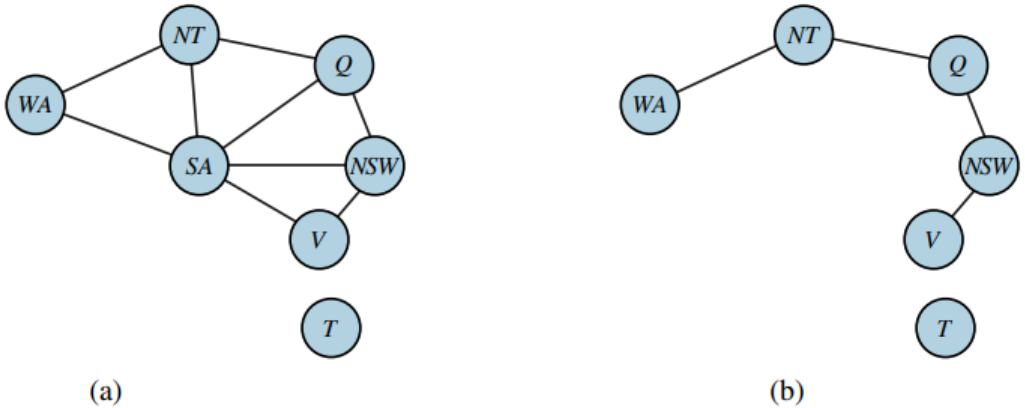


Figure 6.12 (a) The original constraint graph from Figure **??**. (b) After the removal of SA, the constraint graph becomes a forest of two trees.

Dwie wiadomości

- **Zła:** problem jest NP-trudny
- **Dobra:** istnieją heurystyczne rozwiązania, ostatnio pojawiło się używające sieci neuronowych

Przeszukiwanie dla przeszukiwania

Uwaga

Jak mówimy o przeszukiwaniu stanów, to pamiętamy: **stan jest czymś abstrakcyjnym**!

Przeszukiwanie dla przeszukiwania

Uwaga

Jak mówimy o przeszukiwaniu stanów, to pamiętamy: **stan jest czymś abstrakcyjnym**!

Zatem możemy powiedzieć:

- Stanem jest zbiór zmiennych udrzewiający sieć więzów
- Kosztem jest liczba kombinacji wartości tych zmiennych

Przeszukiwanie dla przeszukiwania

Uwaga

Jak mówimy o przeszukiwaniu stanów, to pamiętamy: **stan jest czymś abstrakcyjnym**!

Zatem możemy powiedzieć:

- Stanem jest zbiór zmiennych udrzewiający sieć więzów
- Kosztem jest liczba kombinacji wartości tych zmiennych

Możemy heurystycznie przeszukać tę przestrzeń i następnie uruchomić algorytm rozwiązywania więzów

Bounds propagation/bounds consistency

Uwaga

Czasem dziedziny zmiennej są duże i nie chcemy modelować ich jako zbiorów, a jedynie jako przedziały

Bounds propagation/bounds consistency

Uwaga

Czasem dziedziny zmiennej są duże i nie chcemy modelować ich jako zbiorów, a jedynie jako przedziały

Przykładowo:
Mamy dwie zmienne: $F_1 \in \{0, \ldots, 165\}$ oraz $F_2 \in \{0, \ldots, 385\}$ oraz więz $F_1 + F_2 = 420$

Bounds propagation/bounds consistency

Uwaga

Czasem dziedziny zmiennej są duże i nie chcemy modelować ich jako zbiorów, a jedynie jako przedziały

Przykładowo:
Mamy dwie zmienne: $F_1 \in \{0, \ldots, 165\}$ oraz $F_2 \in \{0, \ldots, 385\}$ oraz więz $F_1 + F_2 = 420$
Można z tego wydedukować: $F_1 \in \{35, \ldots, 165\}$ oraz $F_2 \in \{255, \ldots, 385\}$

Bounds propagation/bounds consistency

Rozumowanie

$F_1 + F_2 \geq 420$

Bounds propagation/bounds consistency

Rozumowanie

$F_1 + F_2 \geq 420$
zatem $F_1 \geq 420 - F_2$,

Bounds propagation/bounds consistency

Rozumowanie

$F_1 + F_2 \geq 420$
zatem $F_1 \geq 420 - F_2$,
czyli $F_1 \geq 420 - \max(F_2)$,

Bounds propagation/bounds consistency

Rozumowanie

$F_1 + F_2 \geq 420$
zatem $F_1 \geq 420 - F_2$,
czyli $F_1 \geq 420 - \max(F_2)$,
czyli $F_1 \geq 35$

Bounds propagation/bounds consistency

Rozumowanie

$F_1 + F_2 \geq 420$
zatem $F_1 \geq 420 - F_2$,
czyli $F_1 \geq 420 - \max(F_2)$,
czyli $F_1 \geq 35$

$F_1 \geq 420 - \max(F_2)$ jest **triggerem**, który uruchamia się podczas propagacji zawsze gdy zmienia się $\max(F_2)$

Bounds propagation/bounds consistency

Rozumowanie

$F_1 + F_2 \geq 420$
zatem $F_1 \geq 420 - F_2$,
czyli $F_1 \geq 420 - \max(F_2)$,
czyli $F_1 \geq 35$

$F_1 \geq 420 - \max(F_2)$ jest **triggerem**, który uruchamia się podczas propagacji zawsze gdy zmienia się $\max(F_2)$

Uwaga

Przykładowy więz był binarny, **ale nie jest to wymagane**!

Łamanie symetrii

Uwaga

Czasami istnieje wiele (równoważnych) rozwiązań, co może niepotrzebnie zwiększać przestrzeń poszukiwań

Łamanie symetrii

Uwaga

Czasami istnieje wiele (równoważnych) rozwiązań, co może niepotrzebnie zwiększać przestrzeń poszukiwań

Przykładowo: dla poprawnego kolorowania możemy spermutować kolory i nadal będziemy mieli poprawne dokorowanie (stąd dla d kolorów jest na pewno $d!$ rozwiązań)

Łamanie symetrii dla Australii

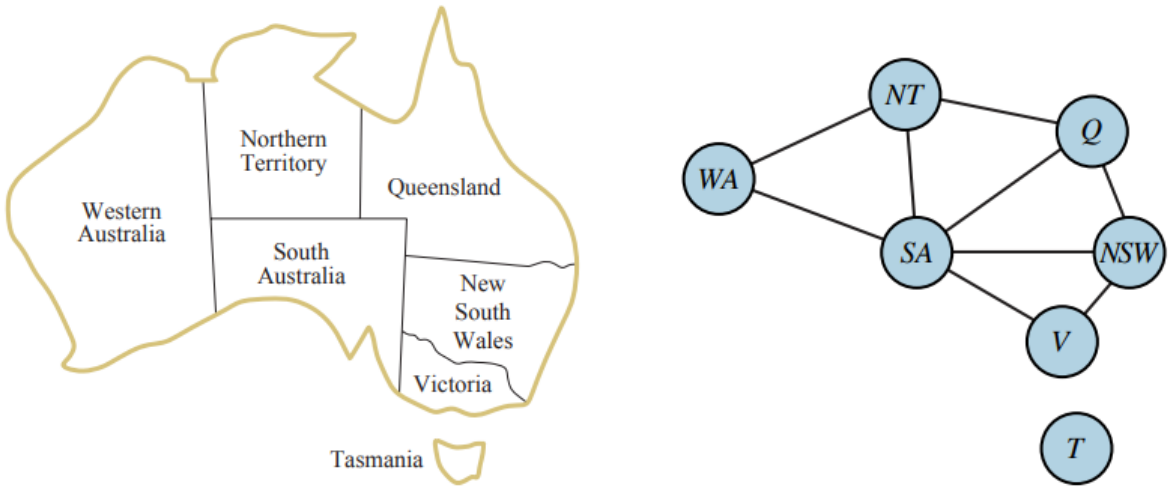


Figure 6.1 (a) The principal states and territories of Australia. Coloring this map can be viewed as a constraint satisfaction problem (CSP). The goal is to assign colors to each region so that no neighboring regions have the same color. (b) The map-coloring problem represented as a constraint graph.

Łamanie symetrii dla Australii

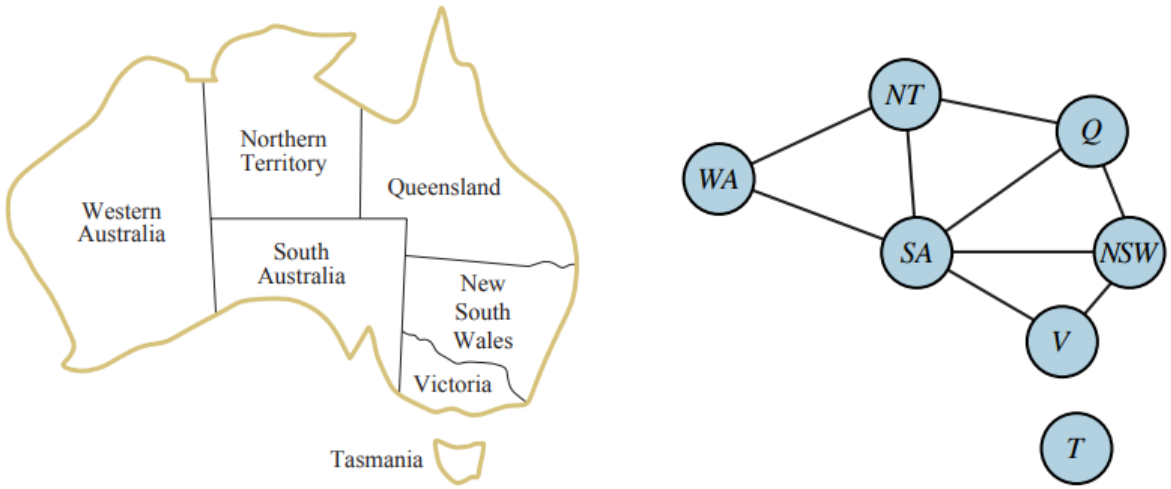


Figure 6.1 (a) The principal states and territories of Australia. Coloring this map can be viewed as a constraint satisfaction problem (CSP). The goal is to assign colors to each region so that no neighboring regions have the same color. (b) The map-coloring problem represented as a constraint graph.

Można dodać więz: $NT < SA < WA$

Drzewo gry:

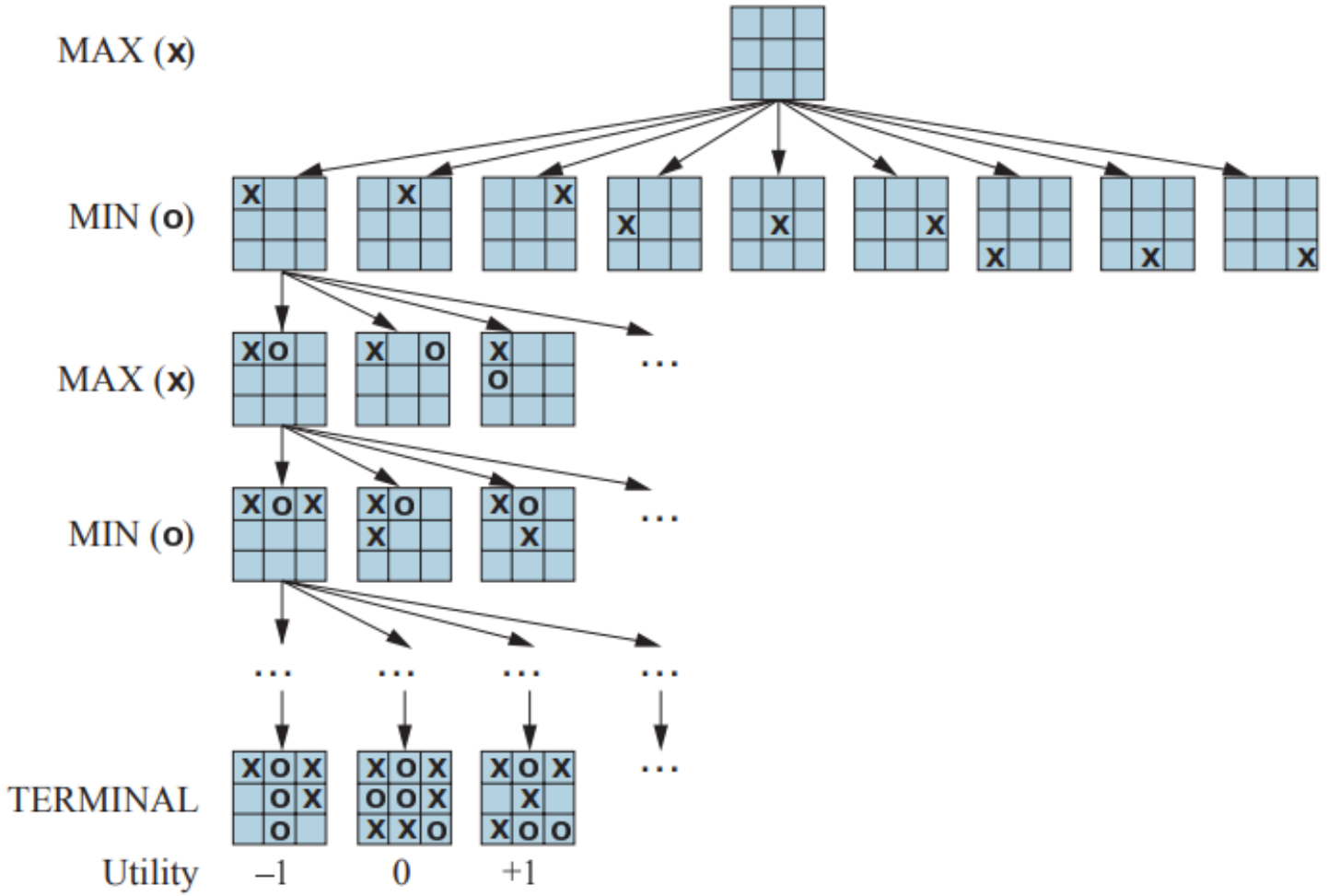


Figure 5.1 A (partial) game tree for the game of tic-tac-toe. The top node is the initial state, and MAX moves first, placing an X in an empty square. We show part of the tree, giving alternating moves by MIN (O) and MAX (X), until we eventually reach terminal states, which can be assigned utilities according to the rules of the game.

O znaczeniu słowa: heurystyka

Definicja (wikipedia)

metoda znajdowania rozwiązań, dla której nie ma gwarancji znalezienia rozwiązania optymalnego, a często nawet prawidłowego

O znaczeniu słowa: heurystyka

Definicja (wikipedia)

metoda znajdowania rozwiązań, dla której nie ma gwarancji znalezienia rozwiązania optymalnego, a często nawet prawidłowego

Bardzo przeciążone słowo w Sztucznej Inteligencji:

1. W algorytmie A^* szacowany koszt dotarcia do celu

O znaczeniu słowa: heurystyka

Definicja (wikipedia)

metoda znajdowania rozwiązań, dla której nie ma gwarancji znalezienia rozwiązania optymalnego, a często nawet prawidłowego

Bardzo przeciążone słowo w Sztucznej Inteligencji:

1. W algorytmie A^* szacowany koszt dotarcia do celu
2. W więzach: First Fail, Least Contstraining Value – heurystyczna metoda wyboru zmiennej i wartości

O znaczeniu słowa: heurystyka

Definicja (wikipedia)

metoda znajdowania rozwiązań, dla której nie ma gwarancji znalezienia rozwiązania optymalnego, a często nawet prawidłowego

Bardzo przeciążone słowo w Sztucznej Inteligencji:

1. W algorytmie A^* szacowany koszt dotarcia do celu
2. W więzach: First Fail, Least Contstraining Value – heurystyczna metoda wyboru zmiennej i wartości
3. W grach: przybliżona ocena sytuacji na planszy

O znaczeniu słowa: heurystyka

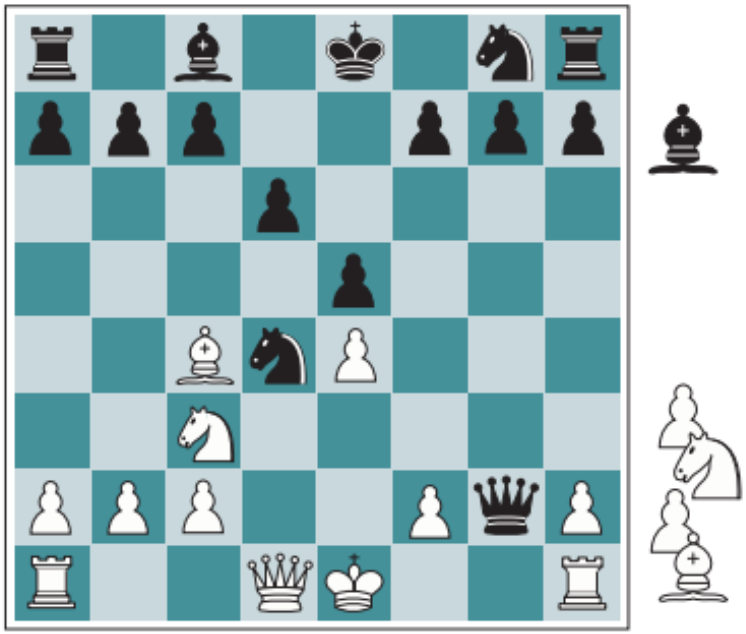
Definicja (wikipedia)

metoda znajdowania rozwiązań, dla której nie ma gwarancji znalezienia rozwiązania optymalnego, a często nawet prawidłowego

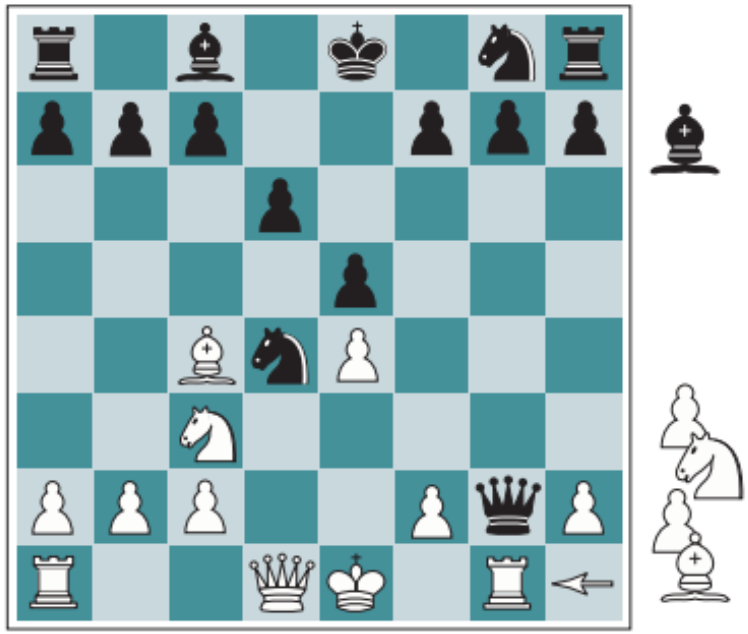
Bardzo przeciążone słowo w Sztucznej Inteligencji:

1. W algorytmie A^* szacowany koszt dotarcia do celu
2. W więzach: First Fail, Least Contstraining Value – heurystyczna metoda wyboru zmiennej i wartości
3. W grach: przybliżona ocena sytuacji na planszy
4. **Algorytmy metaheurystyczne** – rodzina algorytmów związana z przeszukiwaniem (alg. genetyczne, przeszukiwania lokalnego, symulowane wyżarzanie, ...

Ocena sytuacji



(a) White to move



(b) White to move

Figure 5.8 Two chess positions that differ only in the position of the rook at lower right. In (a), Black has an advantage of a knight and two pawns, which should be enough to win the game. In (b), White will capture the queen, giving it an advantage that should be strong enough to win.

Uporządkowanie ruchów w $\alpha\beta$-search

- Lepsze ruchy lepiej analizować najpierw!
- Sortowanie ruchów jest kosztowne (konieczny balans)
- Bardziej opłacalne na początkowym etapie

Uporządkowanie ruchów w $\alpha\beta$-search

- Lepsze ruchy lepiej analizować najpierw!
- Sortowanie ruchów jest kosztowne (konieczny balans)
- Bardziej opłacalne na początkowym etapie

Stosuje się wariant iteracyjnego pogłębiania:

- Posortuj ruchy ze względu na funkcję heurystyczną,
- Wykonaj $\alpha\beta$-search z głębokością 1 korzystając z ustalonego porządku

Uporządkowanie ruchów w $\alpha\beta$-search

- Lepsze ruchy lepiej analizować najpierw!
- Sortowanie ruchów jest kosztowne (konieczny balans)
- Bardziej opłacalne na początkowym etapie

Stosuje się wariant iteracyjnego pogłębiania:

- Posortuj ruchy ze względu na funkcję heurystyczną,
- Wykonaj $\alpha\beta$-search z głębokością 1 korzystając z ustalonego porządku
- Wykonaj $\alpha\beta$-search z głębokością 2 korzystając z ustalonego porządku
- i tak dalej, póki jest czas

Gra końcowa (w szachach)

Końcówki

W grach, w których gracze zbijają swoje bierki, często mamy do czynienia z typowymi **końcówkami**: przykładowo, król i wieża vs król i goniec (4 bierki)

Gra końcowa (w szachach)

Końcówki

W grach, w których gracze zbijają swoje bierki, często mamy do czynienia z typowymi **końcówkami**: przykładowo, król i wieża vs król i goniec (4 bierki)

1. Dla szachów policzono wartości dla **wszystkich** możliwych ustawień **7 lub mniej** bierek
2. (trochę spora: **140 terabajtów**)

Gra końcowa (w szachach)

Końcówki

W grach, w których gracze zbijają swoje bierki, często mamy do czynienia z typowymi **końcówkami**: przykładowo, król i wieża vs król i goniec (4 bierki)

1. Dla szachów policzono wartości dla **wszystkich** możliwych ustawień **7 lub mniej** bierek
2. (trochę spora: **140 terabajtów**)
3. Zrobiono ją dość ciekawie: grając do tyłu

Retrograde

Idea

1. Dla badanego zestawu bierek przejrzyj wszystkie ich ustawienia, zaznacz: **mat w 0 ruchach dla X**
2. Wykonując ruchy do tyłu odkrywaj pozycje, które są wygrane w N ruchach (szczegóły na ćwiczeniach)

Retrograde

Idea

1. Dla badanego zestawu bierek przejrzyj wszystkie ich ustawienia, zaznacz: **mat w 0 ruchach dla X**
2. Wykonując ruchy do tyłu odkrywaj pozycje, które są wygrane w N ruchach (szczegóły na ćwiczeniach)

Ciekawostka

Projekt ten pozwolił znaleźć sekwencję 262 ruchów, które w nieunikniony sposób prowadzą do mata, zawierającą ponad 50-ruchową sekwencję bez bicia i ruchu pionem