Logika obliczeniowa #3
- badanie spełnialności formuł logicznych -

Przygotował:

Dr inż. Jacek Tkacz

Agenda

- Formuła spełnialna
- Formuła niespełnialna
- Tautologia
- Złożoność obliczeniowa
- Drzewo rozkładu formuły logicznej
- Tabelaryczne badanie spełnialności formuły logicznej

Formuła spełnialna

- Jeśli formuła logiczna posiada takie wartościowanie (podstawienie) zmiennych dla których jest ona prawdziwa to formułę taką nazywamy spełnialną
- Problem spełnialności jest rozstrzygalny - można wypróbować wszystkich podstawień których jest 2^N, gdzie N to liczba zmiennych w formule (Metoda ta ma więc złożoność wykładniczą)
- Problem stwierdzania, czy zadana formuła logiczna jest spełnialna, to bardzo ważny ze względów teoretycznych problem teorii złożoności obliczeniowej. W zależności od postaci formuły jest on uważany za problem łatwy (tj. istnieje algorytm wielomianowy pozwalający na jego rozwiązanie) lub trudny (tj. prawdopodobnie algorytm wielomianowy dla niego nie istnieje).

Formuła niespełnialna

- Jeśli nie istnieje takie wartościowanie (podstawienie) zmiennych logicznych dla których formuła jest prawdziwa to formuła jest niespełnialna

Tautologia

- Formuła zawsze prawdziwa
- Dla każdego możliwego podstawienia (wartościowania) jest zawsze prawdziwa
- formuła jest tautologią, gdy jej negacja jest niespełnialna (sprzeczna)

Problem zlezienia wartościowania spełniającego

- Postać dysjunkcyjna (DNF)
  - Problem znajdowania wartościowania spełniającego formuły w postaci DNF jest problemem łatwym, tzn. istnieje algorytm wielomianowy rozwiązujący niniejszy problem
  - Przykład algorytmu można znaleźć w literaturze
- Postać koniunkcyjna (CNF)
  - Problem znajdowania wartościowania formuły w postaci CNF jest problemem trudnym, NP-zupełnym.

Przykładowy algorytm wielomianowy

- algorytm zwraca zbiór zmiennych, którym należy nadać wartość true (pozostałym zmiennym należy nadać wartość false) aby formuła φ była spełniona jeśli formuła jest spełnialna, w przeciwnym przypadku zwraca specjalną wartość null
- formuła φ podana na wejściu jest zbiorem klauzul
- każda klauzula jest zbiorem literałów

```
foreach C_i ∊ Φ {
   ok = true;
   foreach a_j ∊ C_i
      foreach b_k ∊ C_i
         if (a_j==!b_k) ok = false;
   if (ok){
        T=Ø;
        foreach a_j ∊ C_i
          if (a_j jest
   niezanegowaną zmienną x_i)   T =
   T ∪ {x_i};
        return T;
   }
   return null;
}
```

Przykładowy algorytm wielomianowy #2

- Algorytm dla każdej klauzuli sprawdza czy jest ona niesprzeczna. Gdy znajdzie niesprzeczną klauzulę zwraca zbiór zmiennych, które w niej występują jako literały bez negacji
- Algorytm ten nie może posłużyć do rozwiązania NP-zupełnego problemu spełnialnosci formuł w postaci CNF ponieważ w ogólnym przypadku rozmiar formuły przy przekształcaniu jej z postaci CNF do DNF może wzrosnąć wykładniczo

Problem zlezienia wartościowania niespełniającego

- Postać dysjunkcyjna (CNF)
  - Problem znajdowania wartościowania spełniającego formuły w postaci CNF jest problemem łatwym, tzn. istnieje algorytm wielomianowy rozwiązujący niniejszy problem
  - Przykład algorytmu można znaleźć w literaturze
- Postać koniunkcyjna (DNF)
  - Problem znajdowania wartościowania formuły w postaci DNF jest problemem trudnym, NP-zupełnym.

Problem sprawdzania czy formuła jest tautologią

- Jeśli wyrażenie rachunku zdań jest zapisane w CNF, to łatwo (tj. istnieje wielomianowy) sprawdzić czy jest tautologią. Jeśli bowiem istnieje klauzula, która nie zawiera ani stałej prawda ani przynajmniej jednej zmiennej zarówno pozytywnie, jak i negatywnie, to można tak dobrać zmienne, żeby była ona fałszywa - każdej zmiennej występującej pozytywnie przyporządkujemy fałsz, każdej zaś występującej negatywnie prawdę. Wtedy cała CNF nie będzie spełniona, tak więc nie jest on tautologią.
- Jeśli zaś każda klauzula zawiera albo stałą prawda albo przynajmniej jedną zmienną zarówno pozytywnie, jak i negatywnie (w każdym wartościowaniu albo jedna albo druga będzie prawdziwa), to CNF jest tautologią

Problem P i NP

- Problem P – to problem decyzyjny, dla którego rozwiązanie można znaleźć w czasie wielomianowym
- Problem NP – to problem decyzyjny, dla którego rozwiązanie można zweryfikować w czasie wielomianowym
- Każdy problem P jest NP, jednak nie wiadomo, czy każdy problem NP jest P. Jest to jedno z wielkich nierozwiązanych dotychczas zagadnień informatyki.

Problemy NP-zupełne można traktować jako najtrudniejsze problemy klasy NP (z punktu widzenia wielomianowej rozwiązywalności)

Przykład problemu klasy NP

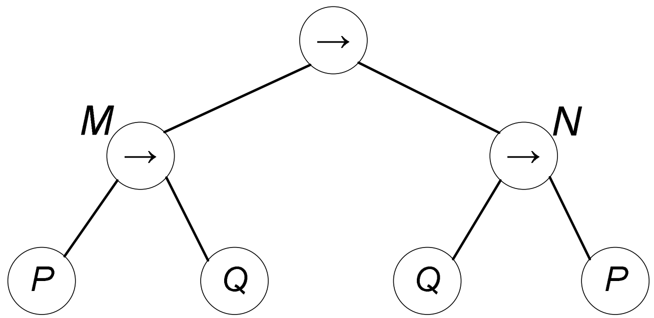
- *Czy jakikolwiek niepusty podzbiór zadanego zbioru (np. {-2,6,-3,72,10,-11}) sumuje się do zera ?*
  - Nasuwający się algorytm sprawdzenia wszystkich możliwych podzbiorów ma złożoność wykładniczą ze względu na liczebność zbioru (nie wiadomo więc, czy problem jest klasy P)
  - uzyskawszy z zewnątrz kandydata na rozwiązanie (np. {-2,6,-3,10,-11}) możemy w liniowym (a zatem wielomianowym) czasie sprawdzić, czy sumuje się do zera. Jest to zatem problem NP.

Badanie spełnialności formuły (1)

- Formuła dwóch zmiennych

$$\big(\underbrace{(P \to Q)}_{M} \to \underbrace{(Q \to P)}_{N}\big)$$

- Drzewo rozkładu



l.p.	P	Q	$P \to Q$ M	$Q \to P$ N	$M \to N$
0	0	0	1	1	1
1	0	1	1	0	0
2	1	0	0	1	1
3	1	1	1	1	1

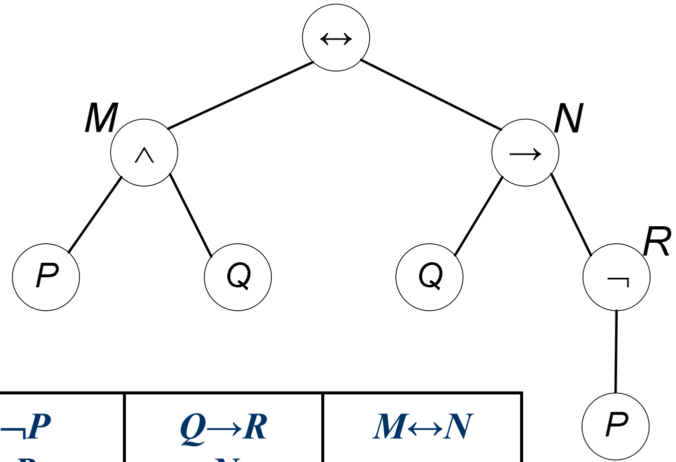
Formuła spełnialna

Badanie spełnialności formuły (2)

- Formuła dwóch zmiennych

$$\big(\underbrace{(P \wedge Q)}_{M} \leftrightarrow \underbrace{(Q \rightarrow \underbrace{\neg P}_{R})}_{N}\big)$$

- Drzewo rozkładu



l.p.	P	Q	$P \wedge Q$ M	$\neg P$ R	$Q \rightarrow R$ N	$M \leftrightarrow N$
0	0	0	0	1	1	0
1	0	1	0	1	1	0
2	1	0	0	0	1	0
3	1	1	1	0	0	0

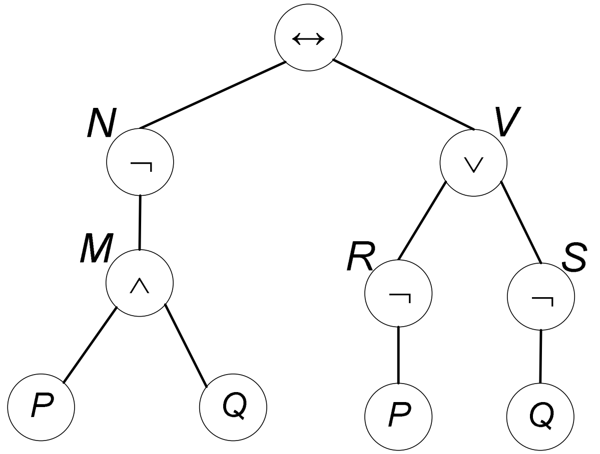
Formuła niespełnialna

Badanie spełnialności formuły (3) – prawo deMorgana

- Formuła dwóch zmiennych

$$\underbrace{(\neg\underbrace{(P \wedge Q)}_{M})}_{N} \leftrightarrow \underbrace{(\underbrace{\neg P}_{R} \vee \underbrace{\neg Q}_{S})}_{V})$$

- Drzewo rozkładu



l.p.	P	Q	$P \wedge Q$ M	$\neg M$ N	$\neg P$ R	$\neg Q$ S	$R \vee S$ V	$N \leftrightarrow V$
0	0	0	0	1	1	1	1	1
1	0	1	0	1	1	0	1	1
2	1	0	0	1	0	1	1	1
3	1	1	1	0	0	0	0	1

Formuła zawsze prawdziwa (tautologia)

Koniec

http://willow.iie.uz.zgora.pl/~jtkacz

Dziękuję za uwagę!