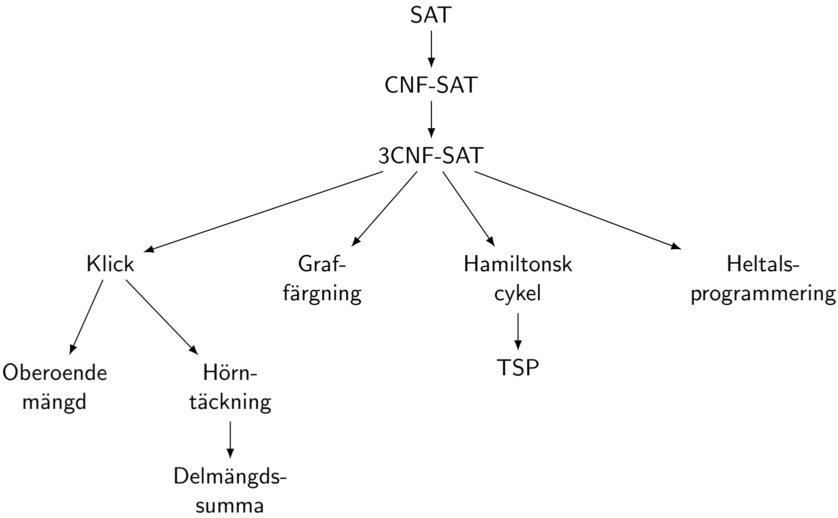
Föreläsning 25 i ADK

NP-fullständighetsbevis

Viggo Kann

KTH

Vidare reduktioner från SAT



Satisfierbarhetsproblem

Allmänt:

- **Indata:** Boolesk formel med booleska variabler
- **Fråga:** Finns det någon variabeltilldelning som satisfierar formeln?

Olika varianter med olika krav på indataformeln:

- **SAT**: Formeln får innehålla operatorerna $\wedge, \vee, \neg, \Rightarrow, \equiv$
  $((x_1 \Rightarrow x_2) \vee (x_1 \vee \overline{x_3 \vee x_4})) \wedge (x_1 \equiv (x_2 \vee x_3))$
- **CNF-SAT**: Formeln måste vara i konjunktiv normalform, dvs vara en konjunktion av disjunktioner av literaler, fär en literal är en variabel eller en negerad variabel.
  $(\overline{x_1} \vee x_2) \wedge (x_1 \vee \overline{x_{17}} \vee \overline{x_{18}}) \wedge x_9$
- **k-CNF-SAT**: Formeln måste vara i CNF och varje disjunktion (klausul) ska bestå av exakt k literaler.
- **3CNF-SAT**:
  $$(x_2 \vee \overline{x_5} \vee x_9) \wedge \underbrace{(\overline{x_1} \vee \overline{x_2} \vee \overline{x_3})} \wedge \underbrace{(\overline{x_1} \vee x_7 \vee x_8)}$$
  literaler klausuler

CNF-SAT är NP-fullständigt

1. Visa att CNF-SAT $\in \mathcal{NP}$

   Bevis:

   $$\left.\begin{array}{l}\text{SAT} \in \mathcal{NP} \\ \text{CNF-SAT} \leqslant_p \text{SAT}\end{array}\right\} \Rightarrow \text{CNF-SAT} \in \mathcal{NP}$$

2. Visa att CNF-SAT är NP-svårt

Bevis 1: Modifiera beviset till Cooks sats (SAT är NP-svårt) och notera att den resulterande formeln kan skrivas om i CNF i polynomisk tid.

Bevis 2: Reducera SAT till CNF-SAT. Givet en SAT-formel φ, konstruera en CNF-formel ψ så att φ är satisfierbar omm ψ är satisfierbar. Detta kan göras genom att man ser formeln φ som en boolesk krets och tillämpar sats 8.13 på sida 471–472 i KT.

3CNF-SAT är NP-fullständigt

1. Visa att 3CNF-SAT $\in \mathcal{NP}$

   Bevis: $\left.\begin{array}{l}\text{SAT} \in \mathcal{NP} \\ \text{3CNF-SAT} \leqslant_p \text{SAT}\end{array}\right\} \Rightarrow \text{3CNF-SAT} \in \mathcal{NP}$

2. Visa att 3CNF-SAT är NP-svårt

   Bevis: Reducera CNF-SAT till 3CNF-SAT

   - Givet en CNF-SAT-formel $\varphi = c_1 \wedge c_2 \wedge \cdots \wedge c_m$ konstruera en 3CNF-SAT-formel ψ som en konjunktion av följande klausuler, för $i \in [1..m]$:
   - Anta att c_i består av j literaler.
     - $j = 3$: använd c_i direkt i ψ
     - $j = 2$: $c_i = (l_1 \vee l_2)$ använd $(l_1 \vee l_2 \vee y) \wedge (l_1 \vee l_2 \vee \overline{y})$
     - $j = 1$: $c_i = (l)$ använd $(l \vee y_1 \vee y_2) \wedge (l \vee y_1 \vee \overline{y_2}) \wedge$
       $\wedge (l \vee \overline{y_1} \vee y_2) \wedge (l \vee \overline{y_1} \vee \overline{y_2})$
     - $j > 3$: $c_i = (l_1 \vee l_2 \vee \cdots \vee l_j)$ använd $(l_1 \vee l_2 \vee y_1) \wedge (\overline{y_1} \vee l_3 \vee y_2) \wedge$
       $\wedge (\overline{y_2} \vee l_4 \vee y_3) \wedge \cdots \wedge (\overline{y_{j-4}} \vee l_{j-2} \vee y_{j-3}) \wedge (\overline{y_{j-3}} \vee l_{j-1} \vee l_j)$

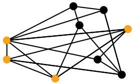
Användbara NP-fullständiga problem 1

- **3CNF-SAT**

Indata: Boolesk 3CNF-formel $(x_2 \vee \overline{x_5} \vee x_9) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_7 \vee x_8)$
Fråga: Finns det någon variabeltilldelning som satisfierar formeln?

- **Clique (klick)**

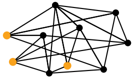
Indata: Oriktad graf G, tal K
Fråga: Finns det K hörn i G som är fullständigt sammanbundna?



$K = 4$ JA

- **Independent set (oberoende mängd)**

Indata: Oriktad graf G, tal K
Fråga: Finns det K hörn i G som är helt oberoende?

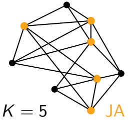


$K = 4$ NEJ

Användbara NP-fullständiga problem 2

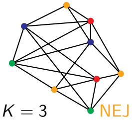
- **Vertex Cover (hörntäckning)**

Indata: Oriktad graf G, tal K
Fråga: Finns det K hörn i G som täcker samtliga kanter?



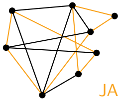
- **Graph coloring (graffärgning)**

Indata: Oriktad graf G, tal K
Fråga: Kan hörnen i G färgas med K färger så att inga närliggande hörn har samma färg?

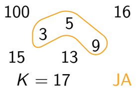


- **Hamiltonsk cykel**

Indata: Oriktad graf G
Fråga: Finns det någon cykel i G som passerar varje hörn i G exakt en gång?



Användbara NP-fullständiga problem 3



- **Subset sum (delmängdssumma)**

Indata: En mängd ickenegativa heltal P, tal K

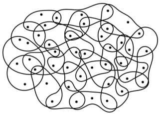
Fråga: Finns det någon delmängd av talen i P vars summa är K?

- **Integer programming (heltalsprogrammering)**

Indata: $m \times n$-matris A, m-vektor b, n-vektor c, tal K. Alla indata är heltal

Fråga: Finns det en n-vektor x med heltal så att $Ax \leqslant b$ och $c \circ x \geqslant K$

$$\begin{pmatrix} 1 & -2 & 3 \\ 0 & 7 & 1 \\ -1 & 2 & -1 \\ 2 & 0 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \leqslant \begin{pmatrix} 0 \\ -2 \\ 1 \\ 3 \end{pmatrix}, \begin{pmatrix} -7 \\ 10 \\ 3 \end{pmatrix} \circ \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \geqslant 47$$



- **Set cover (mängdtäckning)**

Indata: Uppsättning delmängder av en mängd M, tal K

Fråga: Finns det K av delmängderna som tillsammans innehåller alla element i M?

0-1-programmering är NP-fullständigt

1. Visa att 0-1-programmering $\in \mathcal{NP}$
   Bevis:
   - Givet indata A, b, c och K
   - Givet lösning $x \in \{0,1\}^n$
   - Verifiera att $Ax \leqslant b$ och att $c \circ x \geqslant K$
2. Visa att 0-1-programmering är NP-svårt
   Bevis: Reducera 3CNF-SAT till 0-1-programmering
   - Givet en 3CNF-SAT-formel $\varphi = c_1 \wedge \cdots \wedge c_m$
   - Konstruera A,b,c,K så att φ är satisfierbar omm det finns $x \in \{0,1\}^n$ så att $Ax \leqslant b$ och $c \circ x \geqslant K$
   - Inför en variabel x_i för varje variabel y_i i φ
   - Låt $x_i = 0$ motsvara $y_i =$ falsk och $x_i = 1$ motsvara $y_i =$ sant
   - Inför en olikhet för varje klausul c_i

$$\begin{array}{lr} y_i \vee y_j \vee y_k \rightarrow & x_i + x_j + x_k \geqslant 1 \\ y_i \vee y_j \vee \overline{y_k} \rightarrow & x_i + x_j - x_k \geqslant 0 \\ y_i \vee \overline{y_j} \vee \overline{y_k} \rightarrow & x_i - x_j - x_k \geqslant -1 \\ \overline{y_i} \vee \overline{y_j} \vee \overline{y_k} \rightarrow & -x_i - x_j - x_k \geqslant -2 \end{array}$$

   - Låt $c = (0\ 0\ 0\ \ldots\ 0)^\mathsf{T}$ och låt $K = 0$

Exempel

Visa att följande problem är NP-fullständigt

Delgrafsisomorfi
Indata: Oriktade grafer G_1 och G_2
Fråga: Är G_1 en delgraf till G_2?

- Visa att delgrafsisomorfi tillhör $\mathcal{NP}$
- Visa att problemet är NP-svårt genom att reducera ??