3. NPC graph problems

- **P, NP, NPC**
- $\boldsymbol{3SAT \leq IS}$
- $\boldsymbol{3SAT \leq HamPath}$
- $\boldsymbol{HamPath \leq TSP(D)}$
- $\boldsymbol{NAESAT \leq 3 - COL}$

NP er en anden kompleksitetsklasse, som er formelt defineret ved:

$$\forall x \in \{0,1\}^*: x \in L \Leftrightarrow [\exists y \in \{0,1\}^*: |y| \leq p(|x|) \wedge \langle x, y\rangle \in L'] \quad , \quad L' \in \boldsymbol{P} \quad , \quad p\ er\ poly.$$

Dvs. vi vil kunne tjekke om en given løsning er en løsning i polynomiel tid – ydermere må denne løsning maks. have længden polynomisk af probleminstansen.

Dermed kunne vi afgøre om $x \in L$ for $L \in \boldsymbol{NP}$ ved at lave en exhaustive search, hvor der for alle mulige $2^{p(|x|)+1} - 1$ værdier af y tjekkes om $\langle x, y\rangle \in L'$.

NP-hårde sprog er de sprog, der har ret lille sandsynlighed for at ligge i P. Formelt kan vi sige

$$\forall L' \in \boldsymbol{NP}: L' \leq L \Rightarrow L \in NPH$$

NP-hårde sprog ligger ikke nødvendigvis i NP, men de der gør er ret interessante, idet der er en del naturlige problemer, vi ønsker at løse, hvor vi ved, vi ikke kan finde en effektiv løsning (hvis $P \neq NP$). Problemer i både NP og NPH kaldes **NP-complete.**

Lemma 7: Hvis L_1 er NPH og $L_1 \leq L_2$ så er L_2 NPH. **Bevis:** Transitivitet gælder, og idet alle problemer i NP reducerer til L_1 reducerer disse også til L_2, og def. for NPH er opfyldt.

Lemma 7 giver os en anledning til at finde et problem, som er NPC, hvor vi vha. lemma 7 kan reducere andre mulige NPC-problemer til, således vi ikke behøver fører et langt, akavet bevis, hver gang vi vil vise, at et problem er NPC. Cook's sætning gør nøjagtigt dette.

Cook's sætning giver os et problem SAT, som er NP-hårdt (og ligger i NP). Dermed ligger det i NPC. Vi viser at CircuitSAT er NP-hårdt og reducerer til SAT, hvormed Cook's sætning er vist. CircuitSAT generaliserer SAT, idet CNF'er er formularer, som er kredsløb.

3SAT: Vi er givet en 3CNF-formel Φ, og skal bestemme om vi kan finde en sandhedstilordning T som tilfredsstiller Φ. Da 3SAT er NP-complete (og en specialisering af SAT, idet vi reducerer herfra), kan vi bruge 3SAT til at vise, at andre problemer er NPC.

Independent Set: Givet en graf G og et tal K, kan vi finde en uafhængig mængde I i G, således at $|I| \geq K$?

$\boldsymbol{3SAT \leq I.S.}$:

$$(x_1 \vee x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3)$$

Bevis: Først laver vi en oversættelse af 3SAT til I.S. For hver klausul laver vi en trekantgraf, identificer hver knude med en literal. Herefter laves der kanter imellem not og notnot. $K = \#klausuler$.

Knude i I.S. ⇔ tilsvarende literal sand

Korrekthed: ⇒: Antag T opfylder 3SAT-formlen. Vælg en sand literal, som inkluderes i I.S.

⇐: Antag I.S. er en uafhængig mængde af størrelse K. Pga. trekantsgrafen indeholder I.S. præcis én knude fra hver trekant. T er sandhedstilordningen, der sætter literaler svarende til knuder fra I.S. til sand.

Da 3SAT reducerer til I.S., er I.S. NP-hårdt, og da vi nemt kan tjekke en given løsning i polynomiel tid, er det også i NP, dermed er det i NPC.

HamiltonPath: Givet en graf G, kan vi så finde en simpel sti, som besøger hver knude præcis én gang?

$\boldsymbol{3SAT \leq HamPath}$:

Først konstruerer vi en graf ud fra en 3SAT-formel. For hver variabel konstrueres en choice-gadget, som sikrer, at en variabel både kan være sand eller falsk. For hver klausul konstrueres en trekantgraf, hvor hver kant forbindes med en XOR-gadget til en sand eller falsk side på choice-gadgeten for den pågældende literal. XOR-gadgeten sikrer konsistens, idet man kun kan komme tilbage til choice-gadgeten, hvis alle knuder besøges i den. Til sidst laves to knuder, 2 og 3, som hænger sammen, og hvor knude 3 hænger sammen med den sidste choice-gadget.

Sandhedstilordningen T tilfredsstiller Φ ⇔ der er en HamPath i G

Korrekthed: ⇒: Antag T tilfredsstiller Φ.

Løb igennem grafen startende fra 1. Vælg true/false afhængig af T. Herefter løbes resten af grafen blot igennem, indtil man ender i knude 2.

⇐: Antag vi har en ham-path H. Denne starter i 1 og slutter i 2. Definer så $T(x_i) = 1$ hvis H går igennem true i choice-gadget for x_i, ellers er $T(x_i) = 0$. Allerede her er sandhedstilordningen fundet. Resten af ham-pathen interesserer vi os ikke for.

Hvis ikke vi har ramt minimum én kant i hver trekant, efter choice-gadgetene er kørt igennem, så har vi ikke en sandhedstilordning – og dermed ingen ham-path.

TSP: Givet en graf G, kan vi lave en sti, der besøger alle knuder én gang af længde $\leq B$?

$\boldsymbol{HamPath \leq TSP}$:

Givet en graf $G = (V, E)$, som er en instans af ham-path, laver vi en ny graf $G' = r(G)$, med samme knuder som G, men med alle mulige kanter. Der er n byer, en for hver knude i G. d_{ij} er afstanden mellem to byer givet ved:

$$d_{ij} = \begin{cases} 0 & , \ (i,j) \in E \\ 1 & , \ ellers \end{cases}$$

Vi har en tur i G' af længde højst 0 ⇔ G har en HamPath.

Bevis: Vi antager G har en ham-path: Så vil der være en tur i G' af længde højst 0, da hver kant vil være i G. Hvis vi den anden vej rundt, har en tur af længde højst 0, så må hver kant have en cost på 0, og dermed indeholder denne tur kun kanter fra E, og dermed G. Dermed er turen en ham-path i G.

NAESAT: Givet en 3CNF-formel Φ, eksisterer der en sandhedstilordning T, således at for hver klausul, er der mindst en sand og mindst en falsk literal?

3-COLORING: Vi har en graf G, kan vi farve knuderne med farverne {0,1,2} således, to farver ikke rører hinanden?

$\boldsymbol{NAESAT \leq 3-COL}$:

Lav en trekant for hver variabel i formlen, som alle har en fælles knude a – de to andre repræsenterer hhv. not og notnot af variablen. For hver klausul laves en trekant, hvor hver knude repræsenterer en literal, som forbindes til den korresponderende variabel. a-knuden er farvet 2. Farven 1 angiver, at en literal-knude er sand.

Der findes en 3-coloring for G ⇔ der findes en sandhedstilordning T for NAESAT-formlen.

Bevis: ⇒: For hver variabel x_i er den og dens modsætning $\neg x_i$ farvet med to forskellige farver. Sandhedsværdierne for formlen svarer således til farven for x_i.

⇐: Vi farver variable-trekanterne med deres korresponderende sandhedsværdi. I klausul-trekanterne vælges to modsatte sandhedsværdier fra formlen, hvor deres korresponderende knude farves modsat.

Hvis NAESAT-formlen ikke er opfyldt, fx 3 sande literaler, er det ikke muligt at farve klausulgrafen.