

DD1352 Algoritmer, datastrukturer och komplexitet

Övning 10

Anton Grensjo
grensjo@csc.kth.se
<http://csc.kth.se/~grensjo>

24 november 2014

Idag

- Fler bevis av NP-fullständighet

Notera att jag lägger ut dessa anteckningar med reservation för eventuella fel. Hör gärna av er om ni hittar något. Anteckningarna är inte heller fullständiga, utan deras innehåll kompletteras muntligt under övningarna.

Repetition

Definitioner

- Ett problem tillhör NP om det kan verifieras på polynomisk tid.
- Ett problem A är NP-svårt om varje problem i NP kan reduceras till A .
- Ett problem är NP-fullständigt om båda dessa villkor är uppfyllda.

Repetition

Vad bör ingå i ett NP-fullständighetsbevis?

Säg att vi vill visa att problemet A är NP-fullständigt. Då bör vi utföra följande steg:

1. Visa att $A \in NP$.
 - a) Föreslå vad en lösning kan vara.
 - b) Visa att om svaret är ja så kan lösningen verifieras.
 - c) Visa att verifikationen tar polynomisk tid.
2. Visa att A är NP-svårt.
 - a) Hitta ett känt NP-fullständigt problem B att reducera.
 - b) Hitta och beskriv en karp-reduktion av B till A .
 - c) Bevisa att reduktionen är polynomisk.
 - d) Bevisa att reduktionen är korrekt.
 - i) Visa att ja-instanser mappas till ja-instanser och nej-instanser till nej-instanser.

Vi vet sedan att $B \leq_p A$.
3. Nu har vi visat att A är NP-fullständigt!

Konstruera kappsäckslösning

- Se uppgiftslydelse.
- Idé: Använd beslutsproblemet för att avgöra om ett element behövs. Om det inte behövs, släng bort det.
- - 1: **function** KAPPSÄCK(P, S, K)
 - 2: **if** $\neg A(P, S, K)$ **then return** "Ingen lösning"
 - 3: **for** varje föremål $p \in P$ **do**
 - 4: **if** $A(P - \{p\}, S, K)$ **then** $P \leftarrow P - \{p\}$
 - 5: **return** P
- Korrekthet
 - P innehåller inga överflödiga element, ty for-slingan löper över alla element i P och tar bort de som är överflödiga.
 - P innehåller en lösning, ty detta är dels uppfyllt innan slingan, och dels efter varje iteration.

Tillförlitlighet hos Internet

- Se uppgiftslydelse.
- *Beslutsproblem*: Går det att plocka ut en delgraf G' bestående av *exakt* B kanter så att det för alla par av hörn (X, Y) finns minst $T(X, Y)$ disjunkta stigar i G' mellan X och Y ?
- Visa att problemet ligger i NP .
 - Lösning? En delgraf G' enligt ovan. Tillsammans med, för varje par (X, Y) av noder, en lista över minst $T(X, Y)$ disjunkta stigar.
 - Kan på polynomisk tid verifiera genom att
 - a) Kontrollera att G' är en delgraf till grafen.
 - b) Kontrollera att G' har exakt B kanter.
 - c) Kontrollera för varje par av hörn att det krävta antalet disjunkta stigar finns specificerade, är korrekta och disjunkta.

Tillförlitlighet hos Internet

- Visa att problemet är NP-svårt.
 - Vilket problem ska vi reducera?
Hamiltonsk cykel fungerar fint.
 - 1: **function** HAMCYCLE($G = (V, E)$)
 - 2: $B \leftarrow |V|$
 - 3: $T(X, Y) \leftarrow 2, \quad \forall X, Y \in V$
 - 4: **return** INTERNETBESLUT(G, T, B)
 - Reduktionens tidskomplexitet $\in \mathcal{O}(|V|^2)$, polynomiskt i indatastorlek.
 - Korrekthet
 - Antag att vi har en ja-instans s av hamiltonsk cykel, dvs det finns en hamiltoncykel i G . Vi ska då visa att vår konstruerade instans s' av INTERNETBESLUT också är en ja-instans.
Notera att om vi väljer ut delgrafen $G' = (V, E')$, där E precis innehåller kanterna i hamiltoncykeln, så är det en lösning till INTERNETBESLUT. Varför?
 - Antag att vår konstruerade instans s' av INTERNETBESLUT är en ja-instans. Vi vill visa att det existerar en hamiltoncykel i G .
Välj två godtyckliga noder $X, Y \in V$. Det existerar två disjunkta stigar mellan dem. Unionen av dessa stigar är en hamiltoncykel.

Problemet är således NP-fullständigt.

Håstads leksaksaffär

- Se uppgiftslydelse.
- Visa ligger i NP .
- Visa NP-svårt.
 - Vilket problem ska vi reducera?
Vi kan försöka med CNF-SAT.
 - Vi måste visa att varje CNF-formel

$$\varphi = (l_1 \vee l_2 \vee \dots \vee l_i) \wedge (l_j \vee \dots \vee l_k) \wedge \dots \wedge (\dots)$$

med n variabler och m klausuler, som reduceras till en instans γ av LEKSAKER, är satisfierbar om och endast om γ är lösbar.

Håstads leksaksaffär

Reduktionen

Vi reducerar problemet enligt följande:

- 1) Varje variabel x_i motsvarar ett kort c_i .
- 2) Det finns två kolumner med m positioner på varje kort, dvs en rad för varje klausul.
- 3) Om x_i finns med i klausul j , så är det vänstra hålet på rad j i c_i förtäckt.
- 4) Om $\bar{x}_i$ finns med i klausul j , så är det högra hålet på rad j i c_i förtäckt.
- 5) Alla andra positioner har hål.
- 6) Det finns ett extra "magiskt" kort c_{n+1} med hål endast i den vänstra kolumnen.

Antag att φ är satisfierbar. Då löser vi vårt specialfall γ av LEKSAKER genom att låta extrakortet ligga rätt väg, låta c_i ligga rätt väg om $x_i = 1$, men vända på c_i om $\bar{x}_i$. Alla hål täcks eftersom varje klausul är satisfierad.

Omvänt, om γ är löst, välj $x_i = 1$ om c_i ligger rätt väg, $x_i = 0$ om c_i ligger fel väg. OBS: Vad gör vi om extrakortet är vänt?

Processorschemaläggning

- Se uppgiftslydelse.
- Visa ligger i NP.
- Visa NP-svårt.
 - Vilket problem?
 - 3-färgning!

Se Viggos lösning.

Nästa gång

- Approximationsalgoritmer

Jag lägger ut mina presentationer på <http://csc.kth.se/~grensjo>.