Norges teknisk–naturvitenskapelige universitet
Institutt for datateknikk og informasjonsvitenskap

Avsluttende eksamen i
TDT4125 Algoritmekonstruksjon, videregående kurs

Onsdag 27. mai, 2015
Tid: 09:00–13:00

Hjelpemidler: (B) Alle trykte/håndskrevne; spesifikk, enkel kalkulator

Språk: Norsk (bokmål)

Les hele eksamen før du begynner, disponer tiden, og forbered spørsmål til fagstabens hjelperunde. Gjør antakelser der det er nødvendig. Svar kort og konsist. Lange forklaringer som ikke direkte besvarer spørsmålet gis ingen vekt.

Alle deloppgaver teller likt.

Oppgave 1

a) Hva er hovedforskjellen mellom evolusjonær beregning og tradisjonelle tilnærminger til problemløsning innen kunstig intelligens?
Løsning: Her godkjennes alle rimelige karakteriseringer av forskjellene. Eksempelsvar: In traditional AI approaches, a lot of emphasis was put on generating the hypotheses. They were typically conceived by the designers, and subsequently tested. In evolutionary computation you try out a lot of hypotheses, and instead focus your effort on making good evaluations of the different hypotheses. By making good evaluations (i.e. measure of fitness) you can drive the evolution towards finding a solution.

Anta at du har oppgitt et konvekst polygon som en serie med punkter $p_1, \ldots, p_n$, der det er en kant mellom hvert par (p_i, p_{i+1}) og mellom p_n og p_1.

b) Skisser en algoritme som effektivt avgjør om et gitt punkt q er innenfor eller utenfor polygonet. Hva blir kjøretiden?
Løsning: Bruk binærsøk, f.eks. ved å sammenligne vektorene $[p_1, q]$ og $[p_1, p_i]$, for å ende opp med en trekant, og sjekk så i konstant tid. $\Theta(\lg n)$.

I en variant av MAX-3-CNF-SAT skal du ikke maksimere antall termer (*clauses*) som er *sanne*, men antall termer som har minst én sann og én usann literal.

c) Beskriv en randomisert algoritme med forventet approksimeringsratio (*approximation ratio*) $\rho = 4/3$. Forklar hvorfor du får denne ratioen.
Løsning: Mynt og kron, som for MAX-3-CNF-SAT. Relativt rett frem utregning.

En *uavhengig mengde* i en graf er en delmengde av nodene der ingen er naboer. Du ønsker å finne en uavhengig mengde som er så stor som mulig i en gitt graf. Du begrenser deg til grafer der hver node har maksimalt k naboer.

d) Beskriv en approksimeringsalgoritme for problemet, der kostnaden til en løsning er antall noder i den uavhengige mengden. Hva blir kjøretiden? Hva blir ρ? Forklar svaret.
Løsning: Plukk én og én node vilkårlig; for hver node du velger ut, forkast naboene. Dette kan gjøre i lineær tid. For hver node du tar med i løsningen forkaster du maksimalt k noder, så $\rho \leq k + 1$.

Men vi kan være mer presise: I den optimale løsningen må hver av nodene enten være i den approksimerte løsningen *eller* en av naboene, men vi kan ikke ha med begge, så $\rho \leq k$.

I en variant av maks-flyt-problemet (*maximum flow*) har hver kant $(u, v) \in E$ en *nedre grense* $\ell(u, v) \geq 0$ i tillegg til den ordinære kapasiteten $c(u, v)$. Vi krever nå $\ell(u, v) \leq f(u, v) \leq c(u, v)$.

e) Sett opp et lineærprogram for flytproblemet med nedre grenser.
Løsning: Her kan man bruke (eller referere til) programmet i avsnitt 29.2 i læreboka, og bare legge til nedre grenser. Dvs., noe i retning av

$$
\begin{array}{llcll}
\text{Max.} & \sum_{v\in V} f_{sv} & - & \sum_{v\in V} f_{vs} & \\
\text{s.t.} & f_{uv} & \leq & c(u,v) & \forall u,v \in V \;, \\
 & f_{uv} & \geq & \ell(u,v) & \forall u,v \in V \;, \\
\& & \sum_{v\in V} f_{vu} & = & \sum_{v\in V} f_{uv} & \forall u \in V - \{s,t\} \;.
\end{array}
$$

Siden maks-flyt med nedre grenser ikke alltid har en lovlig (*feasible*) løsning, kan vi lage en variant der en kant får lov til å ha flyt 0 som et alternativ til å ligge mellom $\ell(u,v)$ og $c(u,v)$.

f) Sett opp et binærprogram (*0-1 integer program*) for dette nye flytproblemet.
Løsning: Nøkkelen er å bruke en 0-1-variabel x_{uv} til å skru av og på om det går flyt i en kant (u,v), men kan kan ikke multiplisere noen av de andre variablene med denne. I stedet må man skru av og på beskrankningene, f.eks. som dette:

$$
\begin{array}{llcll}
\text{Max.} & \sum_{v\in V} f_{sv} & - & \sum_{v\in V} f_{vs} & \\
\text{s.t.} & f_{uv} & \leq & x_{uv} \cdot c(u,v) & \forall u,v \in V \;, \\
 & f_{uv} & \geq & x_{uv} \cdot \ell(u,v) & \forall u,v \in V \;, \\
 & \sum_{v\in V} f_{vu} & = & \sum_{v\in V} f_{uv} & \forall u \in V - \{s,t\} \;, \\
\& & x_{uv} & \in & \{0,1\} & \forall u,v \in V \;.
\end{array}
$$

Man må multiplisere også $c(u,v)$ med x_{uv} for å unngå å ende opp med $0 < f_{uv} < \ell(u,v)$.

Vi definerer også en *beslutningsversjon* av problemet, der vi ønsker å avgjøre om det finnes en flyt større eller lik en gitt terskelverdi.

g) Bevis at beslutningsversjonen er NP-komplett. Beviset trenger ikke være svært utførlig, men må ha med alle de sentrale komponentene i et slikt bevis. Eventuelle reduksjoner må basere seg på problemer fra pensum.
Løsning: Løsningen er mer utførlig enn det som kreves.

Det er lett å vise at problemet er i NP: Et sertifikat som lar oss verifisere *ja*-svar i polynomisk tid er f.eks. en tilordning av flytverdier til kantene. Vi kan verifisere at beskrankningene er overholdt, og at total flyt er over terskelverdien.

Vi trenger så en polynomisk reduksjon fra et kjent NP-komplett problem, der vi får svaret *ja* på de transformerte instansene av terskelproblemet hvis og bare hvis vi skulle ha hatt *ja* for det kjente NP-komplette problemet. Vi kan f.eks. redusere fra SUBSET-SUM. La oss si at vi har en mengde S med positive heltall, og at vi ønsker å finne en delmengde med sum k. Vi lager oss en kilde s og et sluk t, og en node v_x for hvert tall $x \in S$. I tillegg lager vi en node u.

Vi lar $\ell(s, v_x) = c(s, v_x) = \ell(v_x, u) = c(v_x, u) = x$, og $\ell(u, t) = c(u, t) = k$.

Denne konstruksjonen tar polynomisk (lineær) tid. Så spør vi, finnes det en flyt større eller lik k? Flyten vil jo være enten 0 eller k, og den vil være k hvis og bare hvis denne verdien kan konstrueres som $\sum_{x\in X} c(v_x, u) = \sum_{x\in X} x$ for et subsett $X \subseteq S$, som er det opprinnelige problemet.

h) Beskriv kort en overordnet fremgangsmåte for å løse et slikt problem med *branch and bound*, der du bruker f.eks. Simplex-algoritmen som en subrutine.
Løsning: Bruk *branch and bound* til å sette binærvariablene til 0 eller 1. Gjør om heltallsbegrensningene på de gjenværende binærvariablene til vanlige ulikheter, og løs med Simplex-algoritmen, så får du et optimistisk *bound*.

For en sammenhengende, vektet urettet graf $G = (V, E)$, la den såkalte *kosykelmatroiden* $M^*(G)$ være $(E, \mathcal{I})$, der $\mathcal{I}$ inneholder delmengder av E. Et subsett $I \subseteq E$ er med i $\mathcal{I}$ hvis og bare hvis $(V, E \setminus I)$ er sammenhengende. Vekten til et element $I \in \mathcal{I}$ er summen av kantvektene til I.
Merk: På grunn av en feil, var matroiden I det originale oppgavesettet beskrevet som $(V, \mathcal{I})$. Siden $\mathcal{I}$ inneholder delmengder av E, må matroiden selvsagt være $(E, \mathcal{I})$. På grunn av denne feilen tas oppgavene **i** og **j** samlet ut av sensuren der det er til fordel for kandidaten.

i) Beskriv svært kort en algoritme for å finne et element i $\mathcal{I}$ med maksimal vekt. Hva blir kjøretiden?
Løsning: Komplementet av en basis vil være et spenntre. Komplementet av en maksimal basis vil være et minimalt spenntre. Dette kan finnes med Prim eller Kruskal (med tilhørende kjøretider). Direkte bruk av GREEDY(M, w) fra side 440 vil evt. kreve at man også beskriver hvordan man sjekker om $A \cup \{x\} \in \mathcal{I}$.

j) Bevis at $M^*(G)$ er en matroide.
Hint: Vis at dersom $A, B \in \mathcal{I}$ og $|A| < |B|$ så må $|E \setminus A|$ være strengt større enn $|V|$, noe som igjen har konsekvenser for strukturen til $(V, E \setminus A)$?

Løsning: Fra hintet: $E \setminus B$ er sammenhengende, og må dermed inneholde et spenntre, altså $|V|$ kanter. Siden A er mindre må $E \setminus A$ inneholde flere kanter, og dermed en sykel.

1. E er endelig (fra lærebokas definisjon av en graf).

2. Dersom komplementet av I er sammenhengende, så vil komplementet av ethvert subsett også være det.
3. Anta to mengder $A, B \in I$, der B er større. Vi vet da at $(V, E \setminus A)$ inneholder en sykel. Siden begge komplementene er sammenhengende, så må det finnes en kant e i en sykel i $E \setminus A$ som ikke finnes i $E \setminus B$. Denne kanten er altså i B men ikke i A, dvs., $e \in B \setminus A$. Legger vi e til i A fjerner vi en sykel fra, men komplementet er da fortsatt sammenhengende, så $A \cup \{e\} \in \mathcal{I}$.