Norges teknisk–naturvitenskapelige universitet
Institutt for datateknikk og informasjonsvitenskap

Avsluttende eksamen i
TDT4125 Algoritmekonstruksjon, videregående kurs

Lørdag 24. mai, 2014
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) I en approksimerings-teknikk bygger vi løsninger ved å iterativt øke variablene som representerer brutte beskrankninger (*violated constraints*). Hva kalles denne teknikken?

b) En relasjon i pensum uttrykker forholdet mellom gjennomsnittlig verdi for ikke-negative tilfeldige variable og sannsynlighet for store verdier. Hva heter denne relasjonen?
Løsning: Markovs ulikhet.

c) Anta at du har et problem med et eksponentielt antall begrensninger (*constraints*). Beskriv kort hvordan du ville gå frem for å løse problemet i polynomisk tid, og hva som kreves for at du skal kunne bruke metoden du foreslår.
Løsning: Dersom problemet er et lineært optimeringsproblem og man har et polynomisk beskranknings-orakel kan man bruke ellipsoid-metoden.

d) Beskriv kort og presist hvordan du kan gjøre en randomisert approksimeringsalgoritme deterministisk. Hvilken egenskap er det man vil at resultatet skal ha, og hvordan oppnår man det?
Løsning: Gjør hvert randomisert valg deterministisk, så det maksimerer forventningsverdien. Man sitter da igjen med en deterministisk verdi som er minst like bra som den forventede, og man får et deterministisk bound minst lik det forventede randomiserte.

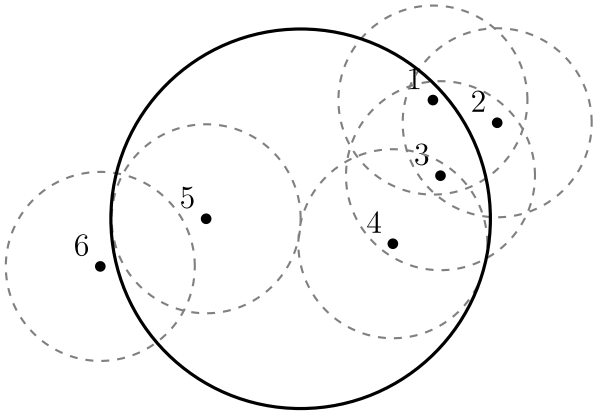
Oppgave 2

a) Fra en gruppe A med n personer skal du velge ut en gruppe B med k representanter til et styre. For hvert par $p = \{x, y\} \subseteq A$ av personer har du estimert en verdi $w(p)$ for hvor enige de er, og dermed hvor godt de kunne representere hverandres meninger. Du ønsker at utvalget skal maksimere summen av disse verdiene. Det vi si, du vil maksimere

$$w(B) = \sum_{p \in (A \setminus B) \times B} w(p)\,.$$

Du hadde til og med vært fornøyd med en algoritme som ga deg et svar som var en konstant faktor unna optimum, men etter en stund innser du at selv dette ville være problematisk. Vis hvorfor det er tilfelle.
Løsning: Reduksjon fra dominating set til approksimeringen. La alle vektene være 0 eller $\ell > 0$. Hvis optimum er 0 finnes det et dominating set av størrelse maks k.

Oppgave 3 Figur 1 viser en partisjonering av en mengde S i hierarkisk snitt-dekomponering (*the hierarchical cut decomposition*). Mengden S er definert av den store, komplette sirkelen; underdelingene (*subdivisions*) er gitt av de mindre, stiplede sirklene.



Figur 1: Hierarchical cut decomposition of a set S.

a) Oppgi delmengdene som S vil partisjoneres i, gitt hver av de følgende tre tilfeldige permutasjonene:

$$\pi_1 = 1, 2, 3, 4, 5, 6$$
$$\pi_2 = 6, 5, 4, 3, 2, 1$$
$$\pi_3 = 3, 2, 1, 4, 5, 6$$

b) Gitt den tilfeldige permutasjonen π_3 fra a) og den resulterende delmengden, hva er størrelsen på den største delmengden gitt i den neste iterasjonen av den hierarkiske snittdekomponeringen? Forklar svaret ditt.

Oppgave 4 Gi algoritmer med polynomisk kjøretid for de følgende problemene:

a) Fargelegging av en tofargbar graf med to farger.

b) Fargelegging av en graf med maksimal grad Δ med $\Delta + 1$ farger.

Oppgave 5 I det urettede maks-snitt-problemet (*maximum cut problem*) har vi oppgitt en urettet graf $G = (V, E)$ og en ikke-negativ vekt w_{ij} for hver kant $\{i, j\} \in E$. Målet er å partisjonere V i to mengder U og $W = V - U$ slik at vi maksimerer den totale vekten på kantene mellom U og W (det vil si, kanter $\{i, j\}$ med $i \in U$ og $j \in W$). Betrakt følgende algoritme for å løse dette problemet: Nummerer nodene $1 \ldots n$. I første iterasjon, legg node 1 i U. I iterasjon k, legg node k i enten U eller W. For å bestemme hvor, se på kantene $F = \{\{k, j\} \in E : 1 \leq j \leq k - 1\}$. Vi velger å plassere node k i U eller W basert på det som vil maksimere antall kanter fra F som er med i snittet.

a) Vis at dette er en 1/2-approksimering.

Løsning: Uvektet versjon: For hver kant i grafen, la den siste noden i ordningen være "ansvarlig" for kanten. La r_i være antall kanter som node i er ansvarlig for. Anta at grafen har m kanter. Vi får da $\sum_{i=1}^{n} r_i = m$. Når vi legger til node i vil vi legge til minst $r_i/2$ kanter i snittet (ellers ville vi ha lagt i i den andre mengden). Vi får dermed minst $\sum_{i=1}^{n} r_i/2 = m/2$ kanter i snittet, og m er helt klart en øvre grense for optimum.

Vektet versjon: La r_i være summen av kantvekter som i er ansvarlig for i stedet for antall kanter. Resonnementet blir det samme.

Merk: Oppgaven, slik den opprinnelig var formulert, var inkonsekvent mtp. bruk av vekting, så studentene får full uttelling enten de har løst den vektede eller uvektede varianten. Dessuten manglet siste setning "som er med i snittet." Oppgaven tas derfor ut av sensur dersom det er til fordel for studenten.

I det rettede maks-snitt-problemet har vi oppgitt en rettet graf $G = (V, A)$ med en ikke-negativ vekt w_{ij} for hver kant $(i, j) \in A$. Målet er å partisjonere V i to mengder U og $W = V - U$ slik at vi maksimerer den totale vekten på kantene fra U til W (det vil si, kantene (i, j) med $i \in U$ og $j \in W$).

b) Vis at problemet kan uttrykkes som et kvadratisk heltallsprogram (*integer quadratic program*). (Hint: Det kan hjelpe å introdusere en variabel y_0 som indikerer om verdien -1 eller 1 betyr at y_i er i mengden U.)

Oppgave 6

a) Vis at den følgende algoritmen for å finne korteste vei fra s til t er ekvivalent med Dijkstras algoritme.

PRIMAL–DUAL SHORTEST s–t PATH ALGORITHM (ALG. 7.4):

$y \leftarrow 0$
$F \leftarrow \emptyset$
while there is no s–t path in (V, F)
  Let C be the connected component of (V, F) containing s
  Increase y_C until there is an edge $e' \in \delta(C)$ such that $\Sigma_{S \in \mathcal{S}: e' \in \delta(S)} y_S = c_{e'}$
  $F \leftarrow F \cup \{e'\}$
end
Let P be an s–t path in (V, F)
return P

Oppgave 7 Vi skal prøve å finne den korteste rundturen (*tour*) i en komplett (urettet) graf. Hver kant $\{i, j\}$ har en positiv (ikke-null) vekt w_{ij}. Betrakt nå følgende to scenarier, for en konstant $k > 0$.

Scenario 1: For enhver sekvens av noder $x, y, u, \ldots, v, z$ holder følgende ulikhet:

$$w_{x,z} \leq k \cdot (w_{xy} + w_{yu} + \cdots + w_{vz})$$

Scenario 2: For enhver sekvens av noder $x, y, u, \ldots, v, z$ holder følgende ulikhet:

$$w_{x,z} \leq k + (w_{xy} + w_{yu} + \cdots + w_{vz})$$

a) Diskuter kort forskjellen mellom scenariene, med tanke på hvor godt du kan approksimere den optimale løsningen.

Løsning: En algoritme som f.eks. Christofides' kan brukes i det første scenariet. Løsningen som består av minimal matching og spenntre vil fortsatt maks være 3OPT/2. Når vi gjør dette til et optimum får vi et sett med "snarveier," koblet i serie med hverandre og evt. deler av den originale node-sekvensen. Dette blir maks k ganger så stort som det vi startet med, så vi kan approksimere til i hvert fall $3k \cdot \mathrm{OPT}/2$.

Scenario 2 er verre. Her kan vi redusere fra Hamilton-sykel-problemet til approksimeringsproblemet vårt (som for generell TSP). For en hvilken som helst ønsket approksimeringsgrad α, sett vekten på de opprinnelige kantene til en verdi mindre enn $1/(\alpha n)$ og vekten til de gjenværende til k. Da er egenskapen opprettholdt, og approksimeringen vil tvinges til å finne en Hamilton-sykel, hvis den eksisterer.