TDT4125 Algoritmekonstruksjon

Eksamen, 18. mai 2022, 15:00–19:00

Faglig kontakt	Magnus Lie Hetland
Hjelpemiddelkode	A

Løsningsforslag

Løsningsforslagene i rødt nedenfor er *eksempler* på svar som vil gi full uttelling. Det vil ofte være helt akseptabelt med mange andre, beslektede svar, spesielt der det bes om en forklaring eller lignende. Om du svarte noe litt annet, betyr ikke det nødvendigvis at du svarte feil!

På grunn av koronapandemien er dette en hjemmeeksamen med alle hjelpemidler tillatt. Følgelig bør ikke eksamenssettet ses som representativt for ordinære eksamener.

6 % **1** Hvilke av følgende utsagn om randomisert avrunding av lineære program stemmer?

- Forventet målverdi blir alltid nøyaktig lik opprinnelig målverdi.
- Restriksjonene tilfredsstilles alltid av forventningsverdiene til variablene.
- Vi kan alltid derandomisere, og få en minst like god gyldig løsning.

Anta at variablene har verdier i området $[0, 1]$, og at dette brukes som sannsynligheten for å runde av til 1.

De første to utsagnene er sanne.

7 % **2** I tabellen nedenfor er det oppgitt flere algoritmer brukt ifm. sideveksling (*the paging problem*). For hver metode, angi om den er (**A**) strengt k-kompetitiv (*strictly k-competitive*), (**B**) online, men ikke kompetitiv eller (**C**) offline.

Metode	**A**	**B**	**C**
FIFO	•		
LIFO		•	
LFU		•	
LRU	•		
FWF	•		
LFD			•
MARK	•		

Her er MARK en vilkårlig algoritme basert på merking (*marking*).

Løsningen er angitt i tabellen.

8 % **3** Plasser riktige ord på riktig plass i teksten nedenfor:

primal, *primal-variabel*, *primal-restriksjon*, *dual*, *dual-variabel*, *dual-restriksjon*

dual, *dual*, *primal*, *primal*

I primal–dual-metoden bruker vi å ha en gyldig *dual*, som vi gjør stadig bedre, og en ugyldig *primal*, som vi gjør stadig gyldigere. Vi sørger også for approksimert komplementær slakkhet. For eksempel, for mengdedekkeproblemet, sørger vi for at hver *dual-restriksjon* er stram før vi runder opp tilhørende *primal*-variabel.

15 % **4** Anta at du skal besøke et sett med planeter (representert ved punkter i rommet), og at du skal besøke hver av dem nøyaktig én gang før du kommer tilbake til startpunktet. Du ønsker å finne en så kort rute som mulig. Hvilken ulikhet bruker pensumalgoritmene her, og hvordan brukes den?

Dette er et spesialtilfelle av metrisk TSP, og vi bruker *trekantulikheten* for å kunne approksimere. (Generell TSP kan ikke approksimeres med mindre P = NP.) For *nearest addition* og *double-tree* brukes trekantulikheten til å garantere at en «snarvei» (der vi hopper over dublett-noder) faktisk er kortere (eller like lang). I Christofides' algoritme brukes den også til vise at en løsning som *kun* inneholder nodene av odde grad i spenntreet *ikke* vil være dyrere enn en løsning som inneholder *alle* nodene.

15 % **5** Én måte å finne en tung bipartitt matching er å grådig velge den tyngste kanten mellom ledige noder så lenge det går. Hvilken ytelsesgaranti (*performance guarantee*) gir dette, som approksimasjon av tyngste bipartitte matching? Forklar og diskuter.

> Vi får $\alpha = 1/2$. Her kan man f.eks. argumentere for at man har snittet av to transversalmatroider (som nevnt i forelesningene; ev. snittet av en hode- og en halepartisjonsmatroide, om man legger på retning), og at snittet av to matroider gir en ytelsesgaranti på $1/2$, men andre argumenter og forklaringer vil også være fullgode.

16 % **6** Du skal bestille et sett med varer fra et sett med nettbutikker. Flere av varene finnes å få kjøpt hos flere av butikkene, så du må bestemme hvilke varer du skal bestille fra hvilke butikker, så det blir så billig som mulig.

Gjør følgende antagelser:

- En gitt vare koster det samme i de ulike butikkene.
- Hver butikk har sin egen fraktpris, som er et fast kronebeløp.

Hvordan vil du angripe problemet? Forklar og diskuter.

Fraktprisen for en gitt butikk er uavhengig av antallet varer som bestilles; så lenge du bestiller minst én vare, må dette beløpet betales.

> Dette er ekvivalent med vektet mengdedekke (*minimum-weight set cover*), der varene du skal bestille er grunnmengden, butikkene representerer mengdene med varer de kan levere, og fraktprisen er vektene.
>
> Her er det naturlig å diskutere måter å løse dette problemet på (f.eks. avrunding, dual avrunding, randomisert avrunding, primal–dual-metoden, grådighet) og at man kan få en logaritmisk ytelsesgaranti.

16 % **7** Du skal finne flervareflyt (*multicommodity flow*), gitt følgende:

- Du skal tilfredsstille etterspørsler (*demands*): $|f_k| = d_k$ for $k = 1 \ldots K$.
- Kildene er identiske: $s_i = s_j$ for $i, j = 1 \ldots K$.

Hvordan vil du angripe dette problemet? Forklar og diskuter.

> Dette kan løses som et vanlig flytproblem med etterspørsler (flyten kan partisjoneres fritt i etterkant), som igjen kan reduseres til vanlig maks-flyt (ved å legge til et nytt sluk t, med kapasiteter $u(t_i, t) = d_i$, som man prøver å fylle opp). Her bør svaret forklare hvordan dette gjøres, og kort antyde hvorfor det blir rett.

16 % **8** For en gitt sammenhengende graf $G = (V, E)$, utgjør sykelfrie delmengder av E en matroide. Hva om vi tillater ikke-sammenhengende grafer, og sier at en mengde er uavhengig mengde dersom den inneholder maksimalt én sykel per sammenhengende komponent? Forklar og diskuter.

Ja, det vil fortsatt være en matroide (en såkalt *bisirkulær* matroide). Hvordan man argumenterer kommer an på hvilken definisjon man bruke for matroider. Det er relativt opplagt at man fortsatt har et uavhengighetssystem. Vi kan så f.eks. bruke den vanlige vekslingsegenskapen (betingelse (2) fra teorem 5.2.1, s. K151), at for ethvert par J, K med uavhengige mengder, der $|J| = |K| + 1$, så finnes det et element $a \in J \setminus K$ slik at $K \cup \{a\}$ også er uavhengig. J kan ikke ha flere sammenhengende komponenter enn K. Hvis den har færre komponenter, må J inneholde en kant mellom to av J sine komponenter, og denne kan trygt brukes som a. Ellers må J ha en sykel i en komponent der K ikke har det. Denne sykelen vil inneholde en trygg a (som enten innfører en sykel i K, eller som kobler sammen to av K sine komponenter).

Dette er en generalisering av en oppgave i øvingsopplegget (første del av oppgave 5.2.5 hos Jungnickel), der man har en sammenhengende graf og dermed tillater maksimalt én sykel totalt.