

Topics in transportation and cargo science

Exercise:

# Implementer en ikke-robust branch-and-cut-and-price algoritme for the Vehicle Routing Problem with Time Windows

Simon Spoorendonk  
DIKU, University of Copenhagen

18. oktober 2005

## 1 Introduktion

Bliv nummer to i verden til at implementere en ikke-robust branch-and-cut-and-price algoritme for the Vehicle Routing Problem with Time Windows (VRPTW).

Til løsning af opgaven henvises især til de slides og undervisningsnoter, som blev benyttet på kurset ved forelæsningsgang 5-8.

Det udleverede rammeprogram er en simpel branch-and-price algoritme til VRPTW, dvs. der tilføjes ingen cuts i master problemet. Derudover er label setting algoritmen til løsning af pricing problemet implementeret ved fuldstændig oprensning af alle veje, dvs. ingen dominanskriterier er tilføjet.

## 2 Opgaven

Opgaven er delt i fire dele:

1. Tilføj dominantstjek i label-setting algoritmen for pricing problemet.
2. Implementer en separationsalgoritme til at udlede overtrådte 2-path uligheder.
3. Implementer en separationsalgoritme til at udlede overtrådte 3-kunde klikker.
4. Modifier dominanskriteriet i label-setting algoritmen for pricing problemet til at behandle 3-kunde klikker.

Udover implementationen forventes en kortfattet rapport, hvor pseudokode/beskrivelse af de implementerede rutiner beskrives.

## 2.1 Godkendelse

Udover løsning af alle fire stillede opgavepunkter, mener den kursusansvarlige, at løsning af følgende kombinationsmuligheder af opgavepunkterne sandsynligvis vil føre til godkendelse af opgaven:

- 1 og 2
- 1 og (3 eller 4)
- 2 og (3 eller 4)

Den opmærksomme læser vil have bemærket, at punkt 3 og 4 ikke giver mening at implementere enkelvis, samt at de afhænger af implementationen af punkt 1.

For ikke at ødelægge humøret for alle (og for at godkendelsespunkterne giver mening), kan kode til punkt 1 og henholdsvis 3 eller 4 blive udleveret ved forespørgsel. Dette vil gøre det muligt at implementere andre, men ikke alle dele af programmet.

Såfremt man ønsker karakter for kurset, kan dette skrives på besvarelsen. Den vil da blive bedømt med intern censur og give karaktergivende ikke-rapportdokumenterede point.

## 2.2 Aflevering

Fredag d. 4 november 11-13 i NO37, hvor der vil være evaluering og resterende ugeopgaver vil blive leveret retur.

## 3 Rammeprogrammet

Rammeprogrammet er implementeret i C++ og bygger på frameworket COIN (<http://www.coin-or.org>). Rammeprogrammet ligger gemt i versionsstyringsværktøjet SubVersion. Der findes en manual på:

<http://svnbook.red-bean.com/>

Tjek koden ud ved at skrive

```
svn co file:///net/urd/home/disk14/spooren/repos/topics426/trunk vrptw
```

Hvis der er problemer med at tilgå `disk14` kan dette klares ved at skrive:

```
ls ~spooren
```

før koden tjekkes ud. Gå nu ind i folderen `vrptw` og opret filen `Makefile.user` og skriv følgende i den:

```
export USERNAME = ditbrugernavn
```

hvor `ditbrugernavn` er dit login. Dette skal bruges til at skrive et unikt sted ned i tempfolderen på den nyværende maskine. Dette nedsætter tiden det tager at linke betragteligt.

Kør da `make dirs` for at lave lidt symbolske links til COIN frameworket, nogle shared libraries samt til probleminstansbiblioteket. Kør nu `make` for at linke.

For at køre programmet skriv da:

```
.\vrp vrp.par
```

hvor `vrp.par` er parameterfilen til at finindstille ting til branch-and-cut-and-price algoritmen. Parametrene i starten af denne fil, dvs. under `USER`, slår kaldene til brugerfunktionerne til og fra. Dette kan være belejligt under udvikling.

### 3.1 Probleminstanser

Parameteret `VRP_Datafile` vælger hvilken instans der arbejdes på. Parameteret `VRP_NumCustomers` vælger hvor mange kunder fra instansen der benyttes.

Probleminstanserne der benyttes er Solomons:

<http://w.cba.neu.edu/~msolomon/problems.htm>

### 3.2 Filer som skal benyttes

I `USER` ligger fire filer. Hver af disse indeholder en brugerfunktion, som kaldes af programmet. Implementer opgavepunkter 1-4 heri, tilføj evt. hjælpefunktioner i disse filer.

Derudover kan man med fordel benytte dokumentationen fra COIN på:

<http://www.coin-or.org/Doxygen/Bcp/hierarchy.html>

for detaljer om fx `BCP_cut` objektet. Der foreligger desværre ingen dokumentation for den udleverede `VRPTW` kode, der må man bare kigge sig frem i koden.

## 4 Spørgsmål

Spørgsmål kan stilles til undertegnede på [spooen@diku.dk](mailto:spooen@diku.dk) eller ved besøg på kontor N323.

## 5 Konkurrencen

Der vil naturligvis være en konkurrence for den flittige studerende. Bedste implementation vil blive belønnet med en øl. Kønne algoritmer belønnes ikke medmindre de er hurtigst i praksis, dvs. med den bedste implementation menes der: Den som løser probleminstanserne hurtigst.

For at deltage i konkurrence skal løsningstider plus skærmudskrift sendes til [spooen@diku.dk](mailto:spooen@diku.dk) inden d. 3 november kl. 24.