Exercises

Lab course Module 3
MAS course
Leuven
May 2013

Cees Witteveen
C.Witteveen@tudelft.nl
Delft University of Technology

Introduction

This manual describes the lab course of Module 3 of the 2012-2013 MAS course at KU Leuven.
The goal of the lab course is to become acquainted with computing schedules and decomposition of scheduling problems as discussed during the lectures. You will be asked to apply several techniques discussed in class.

This lab course consists of 2 parts:

- the first part (assignment 1) is about analyzing properties of Simple Temporal Plans;
  Examples of such plans will be provided
- the second part (assignment 2) is about decomposition of simple temporal plans; you will be asked to find decompositions of such plans and determine the flexibility of such plans.

In order to pass this lab course, you need to submit a single report for both parts combined.
The assignment should be completed in pairs, i.e., two students should work together on a solution and submit a single report.

The deliverables required are:

- Your code produced during the assignment (MATLAB source files).

- Your report of 5 to 10 pages describing your findings, incorporating the questions mentioned in the assignments (pdf file).

Please send your submission to

`c.witteveen@tudelft.nl`

by May 29, 23:59.

What you need for assignment 1

We will use MATLAB and a set of functions. To get started you have to

1. download the file `fastfloyd.zip`. This file is needed to use the Floyd-Warshall all-pairs shortest path computations. Add the function to your MATLAB search path.
2. Download the files `STNExamples.zip`
3. Read the files into matrices encoding distance matrices of the corresponding STNs.

Assignment 1

This exercise is about solving STNs using the FastFloyd routine and a number of routines we require you to write.

1. In the file STNExamples.zip you will find a number of example STNs. We require you to find for all the examples the *earliest starting* and *latest starting* times of the events specified. Note that this might require you to read in the example STN's in the right format as required by the FastFloyd routine.

2. You have to write a subroutine that given an STN and a new arbitrary STN-constraint *c* decides whether or not the resulting STN including the constraint *c* is still consistent, and if so, to update its shortest distance matrix. Provide some test examples of your routine using at least two examples of the STNs provided.

3. Provide a way to construct arbitrary schedules for STN's. You have to construct these schedules *interactively*. That means that you have to write a routine that iteratively
   - asks a user to choose a time point in a given feasible interval for an event,
   - then updates the shortest distance matrix of the STN and
   - then asks for a time point of the next event until a time point has been chosen for all the events.

   We require you to show a test trace of your program using the first STN example (stn49.tab).

4. This exercise is about *schedule monitoring*. Given an STN with time point variables $x_1, x_2, \ldots, x_n$, assume that we have a sequence $s = (a_1, a_2, \ldots, a_n)$ of values for these variables, where a_j is the time value assigned to x_j. We ask you to check whether this sequence constitutes a valid schedule for the STN and, if not, what the *smallest value* a_j in the sequence s that violates some constraint.
   Give a test example of this scheduling monitoring program using example stn145.tab.

5.
The following is a list of planes departing from Amsterdam with a hypothetical time interval for their departure time. There are two runways available. We assume that half of the planes is assigned to runway 1 (the flights at odd positions in the list, such as KL0803, KL0769 etc)) and half of them is assigned to runway number 2 (the flights at an even position in the list, such as KL0577, KL0893).
Still conflicts might occur since planes might want to depart at the same time at the same runway. We define the departure time of a plane as the time the plane is given clearance to enter the departure runway.
We assume that these departure times of the planes assigned to the same runway are ordered according to the order in which they appear in the list; this is an ordering according to their earliest departure time.
In order to satisfy the safety constraints, there should be at least 90 seconds separation time between departures of planes.

It is your task to come up with a most flexible schedule for the departure times of the list of planes satisfying all the constraints. This list is available as `standplan.txt`
You can use MATLAB's LP solver to find a solution.

As a bonus you might try to find a better allocation to runways in order to optimize the flexibility even further.

```
#==================================================
#---- Aircraft list
#--------------------------------------------------
KL0803;Manila;12:00;14:10;F02;B777_200
KL0577;Kano;12:05;14:05;F05;A330_200
KL0769;Bonaire;12:20;14:20;D53;MD_11
KL0893;Shanghai;12:30;14:35;F03;B747_400
KL0663;Houston;12:35;14:40;E02;B747_400
KL6037;Boston;12:59;14:55;E07;A330_200
KL1909;Hanover;13:05;14:10;B22;F50
KL1673;Barcelona;13:05;14:10;D71;B737_800
KL1237;ParisdeGaulle;13:10;14:10;C08;B737_400
KL1115;Stockholm;13:30;14:35;C04;B737_400
KL3157;Dublin;13:35;14:40;D24;B737_800
KL1811;Cologne;13:40;14:10;B27;F50
KL3207;Budapest;13:40;14:45;D18;B737_800
KL1019;LondonHrow;13:45;14:50;D52;B737_800
KL1407;Marseille;13:50;14:50;B28;B737_400
KL1591;Bologna;13:50;14:50;B29;B737_400
KL1189;Bergen;13:50;14:50;C09;B737_400
KL1147;Oslo;13:50;14:50;D63;B737_400
KL1655;Venice;13:50;14:50;D78;B737_400
KL2027;Lyon;13:55;14:55;B25;B737_400
KL1627;MilanMalp.;13:55;14:55;C04;B737_400
KL1285;Edinburgh;13:55;15:00;D05;B737_800
KL1365;Warsaw;13:55;14:55;D06;B737_400
KL1477;Glasgow;13:55;15:00;D49;B737_800
KL3399;Larnaca;13:55;15:00;E06;B737_800
KL1445;Aberdeen;14:00;15:00;D44;B737_400
KL3125;Prague;14:00;15:00;D56;B737_400
KL1311;Toulouse;14:10;14:50;B27;F70
KL1131;Copenhagen;14:10;15:15;C06;B737_800
KL1797;Munich;14:10;15:10;C10;B737_400
KL1931;Geneva;14:10;15:10;D27;B737_400
KL1315;Bordeaux;14:15;14:55;B23;F70
KL1157;Gothenburg;14:15;15:15;D76;B737_400
KL1729;Brussels;14:25;14:55;B26;F50
KL1859;Dusseldorf;14:30;15:00;B24;F50
KL1745;Luxembourg;14:40;15:10;B22;F50
KL1217;Sandefjord;14:50;15:15;B29;F70
#==================================================
```