CS 719 Final Project

Glenn Fung

1 Description of the Project

In this project I made a graphical implementation of the Network simplex method for solve relative small Minimal cost network flow problems (1-10 nodes) and (1-99 arcs). The minimal cost network flow problem P is defined as the problem of finding the flow x_{ij} that solves the following problem:

$$\min_{x_{ij}} \sum_j c_{ij} x_{ij}$$

subject to:

$$\begin{array}{c} \sum_j x_{ij} - \sum_j x_{ji} = s_i, \forall i = 1, \ldots, N \\ 0 \leq x_{ij} \leq b_{ij}. \end{array}$$

where N is the number of nodes in the Network, s_i are the divergences or supplies in each node i, c_{ij} are the costs of each arc ij and b_{ij} are the capacities in each arc ij.

The interface allows to the user to introduce the problem data with a minimum use of the keyboard and tries to show all the information corresponding to the resulting trees in each iteration in a easy and understandable way. In order to find a feasible starting tree the big M method is used. At the beginning of the run the related Big M problem is shown and the original arcs are represented with blue arrows and the artificial arcs are represented with grey arrows.

In each iteration the corresponding Basic tree is showed and the arcs are described as follows:

- **Original arcs** are represented by a **blue** arrow.
- **Artificial arcs** are represented by a **grey** arrow.
- The **in-arc** is represented by a **yellow** arrow.
- The **out-arc** is represented by a **red dashed** arrow.
- The arcs in the **cycle** are represented by a **red dotted** arrow.
- Combinations of above representations are valid. In example: a **red dotted** and **yellow** arrow represents the arc that is entering to the base (**in-arc**) to form the **cycle**.

The program was written in MATLAB using the GUI (Graphical User Interface) and consist in a set of "small" subroutines in the form of .M script files and .M function files.

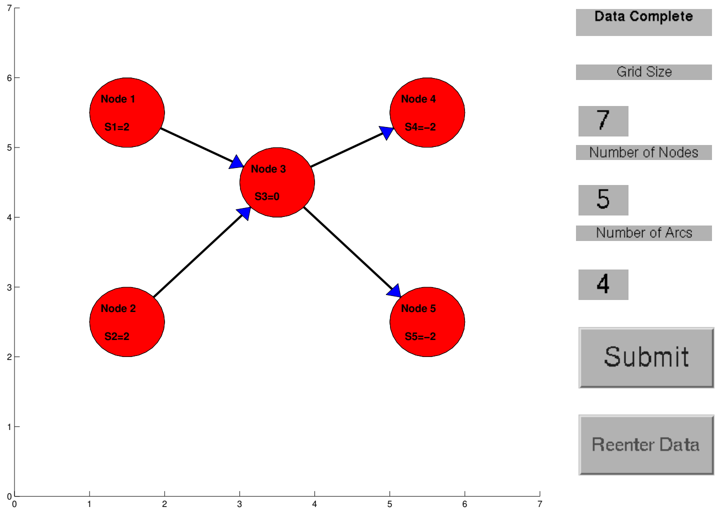

Figure 1: Sample of Input Screen

2 Brief Description of the program

To run the program is necessary to have all the .M files in the same directory and from the MATLAB prompt sign ($>>$) the name of the main program: **mincost** must be typed.

The program has two principal parts:

1. The Input screen.
2. The Network Simplex Method Screen.

2.1 The input screen

In this part of the program the user enters the data in the computer in the following steps:

- Enters the Grid size (Using the mouse).
- Enters the Number of Nodes (Using the mouse).

Enters the Nodes by clicking in the place where the node will be and then Inputs the corresponding divergence value in each node. Note that in this problems, in order to have feasibility we need that the following condition holds:

$$\sum s_i = 0$$

If this condition is not satisfied, the program will show a "Problem not Feasible" message and the program will stop.

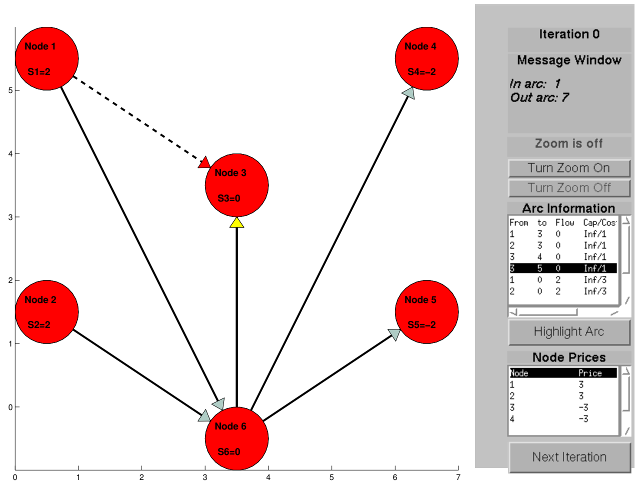

Figure 2: Sample of Network Simplex Method Screen

- Introduces the Number of Arcs.
- Introduces the arcs by clicking first in the starting node of the arc and then in the ending node of the arc. After introduce each arc the user will be asked to enter the Capacity and the Cost corresponding to this arc using the format "capacity/cost", for example, if the user wants to introduce an arc with capacity 3 and cost 5, 3/5 must be introduced.

After all the previous steps the user have to choice between two buttons in the screen, one is the "Submit" button, for solve the problem or the "Reenter data" button for enter all the data again.

2.2 The Network Simplex Method screen

This part of the program consist in the following steps:

1. **STEP 1:** The big M model is showed and a window with all the problem data is also showed, the user can highlight any arc by touching the corresponding data in the window. The program waits until the next iteration (iteration 0) is required by the user.
2. **STEP 2:** The starting feasible tree is showed, the incoming and outgoing arc are showed and the program waits until the next iteration is required. this step is repeated until one of the following three states is reached:
    - Optimality : Optimality of the Big M problem reached and all the artificial arcs have 0 Flow.

- Infeasibility: Optimal of the Big M problem reached but some artificial arc has non zero flow.
- Unboundeness of the Big M problem: in this case, the original problem can be either unfeasible or unbounded.

3 Conclusions

Although the program can considerably be improved, I think it will bring one valid option to introduce students to learn how the Network simplex method works. Also it would be a useful tool if it is used with a computer projection system showing Network simplex method examples in “real time” in a CS719 practice session.