Maximum Flow Problem

João Pedro Vasconcelos Teixeira
Computer Engineering
Informatics Center, UFPB
João Pessoa, Paraíba
Email: joaoteixeira@eng.ci.ufpb.br
Github: https://github.com/jpvt

João Wallace Lucena Lins
Computer Engineering
Informatics Center, UFPB
João Pessoa, Paraíba
Email: jwallace.lucena@gmail.com
Github: https://github.com/joallace

Itamar de Paiva Rocha Filho
Computer Engineering
Informatics Center, UFPB
João Pessoa, Paraíba
Email: itamardprf@gmail.com
GitHub: https://github.com/ItamarRocha

***Abstract*—In this article, we will discuss and expose the results of our first assignment of the Operations Research Course at the Federal University of Paraíba. In this project, we implemented, in C++, Julia, and Python, a modeling of the maximum flow problem, which is one of the most classic and important cases of the minimum-cost flow problem.**

I. Introduction

First of all, it is necessary to present the acronyms that will be used during this report.

OP	Operations research
MFP	Maximum Flow Problem
MCFP	Minimum-cost flow problem

In this coursework, we will implement a solver of the MFP through a modeling of the MCFP. We will execute the project in three programming languages, C++, Python and Julia, with the aid of third party packages, the IBM's CPLEX, Google's OR-Tools, Gurobi and GLPK, in order to compare the performance and usage of each tool and for a better comprehension of the problem.

II. Problem definition

The MFP consists of achieving the maximum possible feasible flow through a network from a specified source node to a specified sink node without exceeding the capacity of the edges that connect those nodes. The MFP is commonly used as a base for different applications in diverse sectors such as distribution planning and scheduling.

In figure 01 we can see a network that we will use to exemplify the MFP, where the numbers in the nodes represent their names and the numbers near the edges represent their capacity.

In this figure we can observe that the maximum flow in the graph (using node 0 as source node, node 5 as sink node and all other nodes as transshipment nodes) is 23.

In general terms, the MFP can be described as follows:

1. All flow through a directed and connected network originates at one node, called **source**, and ends at the node **sink**.

2. All the other nodes are *transshipment nodes*.

3. Flow through an arc is indicated by the arrowhead, where the maximum amount of flow is given by the *capacity* of that arc. At the *source*, all arcs point away from the node. At the *sink*, all arcs point into the node.

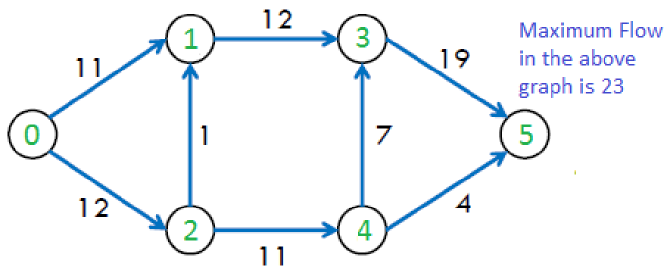


Fig. 1. Visual representation of the Network

4. The objective is to maximize the amount of flow from the source to the sink. This amount is measured by the amount *leaving the source* or *entering the sink*.

III. Modeling

As explained before, the MFP is a special case of the MCFP. Therefore, we will do the modeling of the MFP as an MCFP. The Decision Variable we will be using in our modelling is x_{ij}. This variable represents the used capacity of the edge that connects the node i to the node j.

As we are modeling the max flow problem as the min-cost flow problem, the nodes

N = nodes
x_{ij} = the flow in edge i $\rightarrow$ j
c_{ij} = cost of the edge i $\rightarrow$ j
u_{ij} = capacity of the edge i $\rightarrow$ j
b_i = demand of the node i

Objective Function

$$\text{Min} \sum_{i \in N} \sum_{j \in N} x_{ij} * c_{ij}$$

Our Objective Function will aim to minimize the overall cost. As we have said before, the only edges that have a cost are the ones that we created. Therefore, all the other original edge's costs will be zero.

Constraints

Here we will describe the constraints that we used to model our problem.

• **Capacity constraint**

$$0 \leq x_{ij} \leq u_{ij}$$

The flow in each node must not exceed it's maximum capacity

• **Flow Conservation Constraint**

$$\sum_{j \in N} x_{ij} - x_{ji} = b_i, \forall i \in N$$

the sum of the edges flow that enters the node minus the ones that leave must be equal to its demand

To get the max flow of the network, we had to do the following adaptations:

• Create new edges that connect the starting nodes to the end node and assign a capacity equal to the max capacity that could flow from it's starting node. As an example:

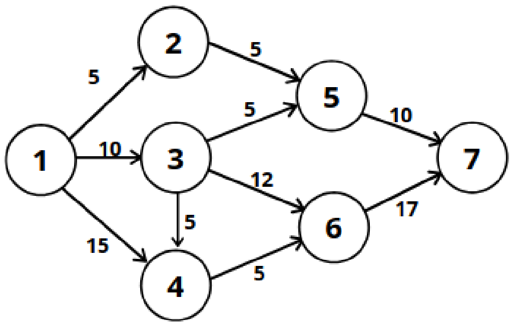


Fig. 2. Network example

In the case of the figure 02, the max capacity that we refered would be the sum of the capacities of each edge that receives flow from the source. Taking this into consideration, the max flow that we will use in the edge we will create in this case is going to be u_{12} + u_{13} + u_{14} = 30. Resulting in graph of figure 03.

• The cost variable c_{ij} (that represents the cost in each edge) was initialized with zero to all edges and then assigned 1 to all edges that we created.

• All the transshipment nodes will have a demand equal to zero, while the source nodes will have their offer equal to the sum of their outputs maximum capacity, and the sink node will have a demand equal to the sum of the source nodes offer, multiplied by minus one.

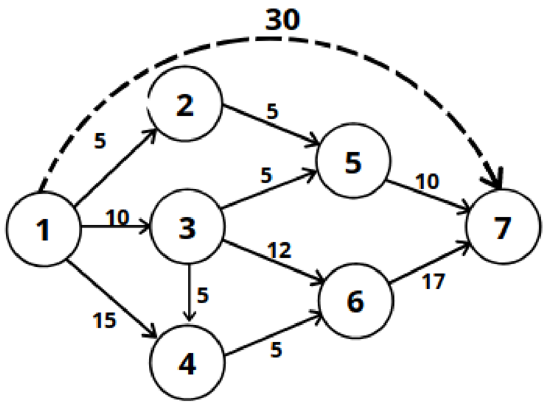


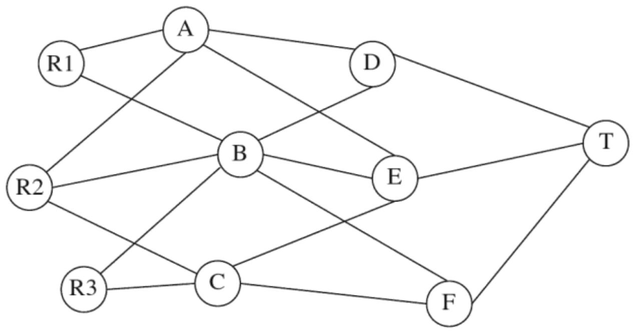
Fig. 3. Created Edge

With the adaptations above, we will be able to discover the max flow of the network, since only the flow that cant go through the original network will be able to pass through the created edge. We can determine the max flow by subtracting the result of the objective function (which will correspond to the flow through the created edge) from the max capacity that we previously cited.

IV. Exercise Solution

We've been assigned to solve the following question, from the book *Introduction to Operations Research* by Hillier and Lieberman:

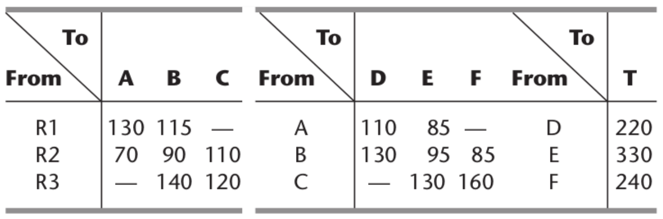
Question. The next diagram depicts a system of aqueducts that originate at three rivers (nodes R1, R2, and R3) and terminate at a major city (node T), where the other nodes are junction points in the system.



Using units of thousands of acre feet, the tables below the diagram show the maximum amount of water that can be pumped through each aqueduct per day.

The city water manager wants to determine a flow plan that will maximize the flow of water to the city.

a) Formulate this problem as a maximum flow problem by identifying a source, a sink, and the transshipment nodes, and then drawing the complete network that shows the capacity



From \ To	A	B	C
R1	130	115	—
R2	70	90	110
R3	—	140	120

From \ To	D	E	F
A	110	85	—
B	130	95	85
C	—	130	160

From \ To	T
D	220
E	330
F	240

of each arc.

Solution. Since this problem has multiple source nodes, we will have to determine them in our code, and we did that by analyzing each node's input and output capacity, with the nodes that do not have an input being the starting ones. After that, we can calculate $\bar{F}$, i.e. a safe upper flow limit, by summing all the source nodes output capacity. Then, we just need to add the imaginary edges connecting the sink node to all source nodes, and we are ready to solve the problem just as any other MFP with one source node.

Answer. The maximum amount of water that can be pumped through each aqueduct per day is **715** units of thousands of acre feet,

To solve this problem, we used different solvers and compared them for learning purposes and to get the most efficient solution. In the following table, we compared the four solvers average execution time, using time in seconds as the evaluation measure, and highlighted the best solution for this problem.

C++ CPLEX	C++ OR-Tools	**Python Gurobi**	Julia GLPK
0.0006	0.0025	**0.0004**	0.0009

Finally, it is noticeable that Gurobi was the best solver for this problem with 0.0004 seconds of execution time.

Although, it is important to say that our main goal was to compare the solvers, which are all implemented in C and C++. With that said, we didn't take into consideration the time for reading the files and other operations.

References

[1] Ahuja R.K., Magnanti T.L., Orlin J.B. (2001) Maximum Flow Problem. In: Floudas C.A., Pardalos P.M. (eds) Encyclopedia of Optimization. Springer, Boston, MA

[2] Frederick S. Hillier, Gerald J. Lieberman - Introduction to Operations Research (2010, McGraw-Hill)