Optimal investment level for increase the capacity of an transportation network

Dorel Dusmanescu, Petroleum and Gas University of Ploiesti, Romania, doreld@upg-ploiesti.ro
Daniela Enachescu, Petroleum and Gas University of Ploiesti, Romania, denachescu@upg-ploiesti.ro
Mirela Matei, Petroleum and Gas University of Ploiesti, Romania, mirematei@yahoo.com

Abstract
In this paper we presents an algorithm for minimizing the investments for increase the capacity of a transportation network. A transportation network is a graph that simulates a real network like roads network of a country, pipes network for fluid transportation etc. In this paper we consider the case when we need to increase the transportation capacity of the network with a specified quantity from the substance transported. The problem was previously treated in [1] for the case of a simple transportation network. In real life we work with complex network, with many inputs and many outputs (i.e. transportation network for water or gas distributions etc.). When we need to increase the capacity of the network we have two solutions: increase the capacity of one or many existing arcs or installing a new arc between two nodes. This problem involves a total cost of the operations that depend on the selected solution. The aim of this work is to refine the algorithm from [1] for increasing the capacity of a transportation network with a determined value including the internal requests and restriction of the network.

Key-Words: - transport network, arc, node, flux, capacity

1. Introduction

A transportation network is an oriented graph that simulate real network that facilitate the transport of something (substance, cars, peoples etc.) between two or many points.

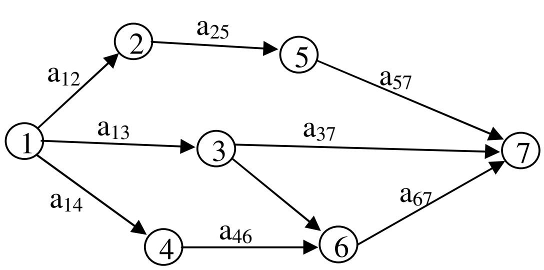

Fig. 1. Transportation network

The network is formed by nodes and oriented arcs. By every arc flows a quantity of substance or a number of cars in every time unit. Every arc has a limited capacity of transportation expressed in flux unit.

The node 1 is the inner node (source) for the network and the node 7 is the outer node (destination or sink). We have some arcs noted in the figure 1 with a_{ij} where *i* and *j* represent the node *i* and *j*.

The capacity of every arc is noted with c_{ij} and represents the maximum flux by the arc. The flux from the network need to respect the next rules (similar to Kirkhoff laws for electrical domaine): in every node the flux that input is equal with the flux that output.

The entire network has a maximum capacity that can be determined with one of these methods: dinamic programming, Ford-Fulkerson method and preflux method.

If we need to increase the capacity of the network we can generate a solution analyzing the network configuration.

But real network use arcs that have specified length and the technical operations for increase the capacity of the arc determine a specified cost. Supplementary, a real network induce some constraints in term of specific requests and restrictions.

For a real network used for water distribution in a city we have a lot number (>1) of sources (considered entry points for the network) and a great number of outer nodes represented by the consumers (figure 2).

This network can be solved in the same mode like clasical network inserting two virtual nodes for start and destination.

For increase the capacity of transportation network we need to know what outer nodes need an increased quantity of substance transported and what are the arcs with restrictions.

In real network the solution for increasing the capacity of the network need to consider the characteristics of the entire network and the charasteristics of subnetwork.

2. Problem Formulation

We suppose that we have a transportation network that simulates a real network like a pipes network for fluid transportation, a roads network from a city or from a country, an electrical network for electrical signals transfer etc. (fig. 2).

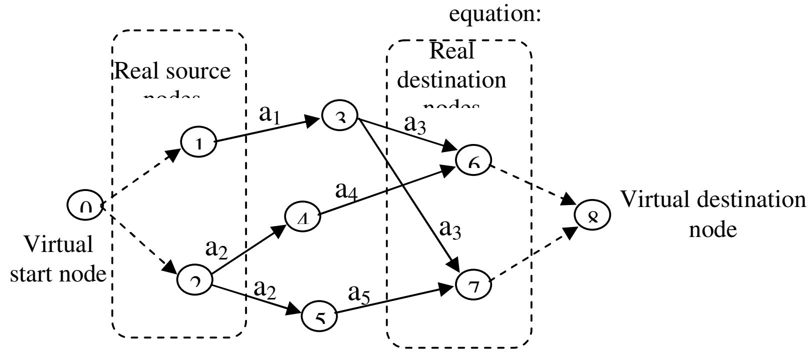

Fig. 2. Complex transportation network with multiple inputs / outputs

In real life appear situation that impose to increase the network capacity with a specified value of flux, value distributed between some nodes, supporting specific restriction and obtaining a minimum cost for the investments.

Because every arc had a correspondent in the real world we need to consider the necessary cost for increasing the every arc capacity.

But every arc had a specified length and a specified cost for increase his capacity.

In case of real correspondent of the arcs we analyze roads or pipes or other structures. To extend the capacity of a road is necessary to extend the width of the road. To extend the capacity of a pipe is necessary to replace the pipe with other, with a large diameter.

These operations involve many costs that can not be ignored if we need to increase the transportation network capacity.

For this reason we need to consider these costs when we want to solve the problem of increasing the network capacity.

The total cost of operations depends by the next factors:
- the arcs combination selected for increasing capacity;
- the length of every arc selected;
- the cost of operations for every arc selected.

Because the physical characteristics of the arcs could be differed for every arc we consider the cost per length unit and per flux unit, and we note this cost with CU_{ij}. This is the cost necessar to increase the flux with one unit in an arc with one unit length.

The entire cost for increasing the capacity of arc a_{ij} depend the difference between the final capacity of the arc and the initial capacity.

If we note this difference with Δ_{ij} we obtain the equation:

$$CA_{ij} = L_{ij} \times \Delta_{ij} \times CU_{ij} \quad (1)$$

where L_{ij} is the length of arc a_{ij},
CA_{ij} – the entire cost for the a_{ij} arc.

For entire network the cost of operation is:

$$C = CA_{ij}+CA_{kp}+CA_{ln}+ \dots +CA_{qr} \quad (2)$$

We can observe, from equation (2), that the arcs that induce the increase of the entire network capacity can not be determined using specified rules.

For a complex transportation network we can not use a deterministic algorithm for selecting the arcs that minimize the cost of increasing the transportation network capacity. The only possibility is to use heuristic algorithms.

Supplementary, in a real network for fluid distribution the need for increased capacity are local, not global (for adding new consumers or for increasing the fluid quantity used by some existent consumers), and the technical possibilities to increase the capacity of the arcs are restricted for some arcs because technical or economical reasons.

3. Problem Solution

This problem are the main characteristics of an Greedy type problems.

Greedy algorithms are simple and are used to obtain the solution for optimisation problems like the minimum length route for a graph, the minimum time of attendance etc.

The solution for a greedy optimization problem consists by a combination of a lot number of elements.

These problems consist of:
- a lot number of candidates for solution;
- a function that verify if a set of candidates could be a problem solution, not necessary optimal;

- a function that verify if a solution set could be completed with another candidate and the resulted set still be a solution for the problem;
- a selection function that permit to select the optimum candidate from the unused candidates;
- a function purpose that offer the value of one solution and represent the function that can need to be optimized.

A greedy algorithm constructs the solution step by step. At the beginning the candidate set is blankness. At every step the algorithm try to append to this set the optimum candidate from unused candidates using selection function.

If, after the append of the last candidate, the set is not acceptable the last candidate is eliminated and will not be considered never. If, after the append, the set of candidates is acceptable the process continue with the next step.

Every time when are extended the candidates set we verify if this set consist a solution for the problem.

If greedy algorithm works correct, the first solution accepted will be an optimal solution for the problem. The optimal solution is not necessary unique. Could be possible that objective function have the same optimal solution for more solution sets.

Formal description of a general greedy algorithm is presented to the next:

function *greedy*(*C*)
{*C* is the multitude of candidates }
S ← Ø {*S* is the solution set}
while not *solution*(*S*) **and** *C* ≠ Ø **do**
{ *x* ← element from *C* who minimizes the function *select*(*x*)
C ← *C* \ {*x*}
if *acceptable*(*S* ∪ {*x*}) **then** *S* ← *S* ∪ {*x*}
}
if solution(*S*) **then return** *S*
else return "do not exists solution"

A greedy algorithm doesn't obtain every time the optimal solution or a solution. It is only a general method that will be necessary in every case to establish if the solution is optimum or not.
In the case of a transportation network we need to establish the functions for verifying if an arc could be into solution set and if the selected set is an acceptable solution.

An arc could be in one of this situation:
- the capacity could be increased with a Δ_{ij} value, determined by the flux quantity that input in the *i* node and the flux quantity that output from the *j* node;
- the capacity could be increased with a value less than Δ_{ij} for technical reasons;
- the capacity of the arc could not be increased for technical reasons;
- the final cost of investments for the arc exceed an specified amount.

For an arc a_{ij}, the Δ_{ij} value represents the absolute minim value for the difference between the arc capacity and the flux that input tot the inner node of the arc (the node *i*) and the difference between the arc capacity and the flux that output from the outer node (the node *j*).

For example, considering the network from fig. 1, the Δ_{25} value is the minim value of the next differences:

$$d_1 = c_{12} - c_{23} \tag{3}$$

$$d_2 = c_{57} - c_{23} \tag{4}$$

$$\Delta_{25} = \min(d_1, d_2) \tag{5}$$

It is logical because if the arc a_{23} is saturated and we want to increase his capacity we can not exceed the flux that input in the node 2 or the flux that output from the node 5. We can increase the capacity of the arc a_{23} until the flux in the arc is equal with the flux that input in the node 2 or with the flux that output from the node 5.

If the nodes are without ramifications the capacity of the arc can be increased until one of her neighbour arcs are saturated.

From Ford-Fulkerson method we know that the network maxim capacity is obtained when a number of arcs are saturated (i.e. the flux is equal with the arc capacity) and these arcs form one cut of network. A cut of network is a set of saturated arcs that can not be eliminated from the network because the results consist in the disconnecting of the destination mode from the source.

It's obviously that the first arcs that can determine an increasing of the transportation network capacity are the saturated arcs. For this reason we consider these arcs for solving the problem.

The saturated arc can be part of the cut of the network or can be only saturated arc.

But only saturation is not sufficient for selecting an arc for increasing her capacity.

The position is almost important for selecting one arc (without any informations about the cost of operation for increase the capacity of the arc).

Solving the network with Ford-Fulkerson method we obtain a preliminary list of saturated arcs.
With this list we start to solve the problem calculating the Δ_{ij} for every saturated arcs and the cost for increasing every saturated arcs with the calculated value Δ_{ij}. For every arcs we verify if exist resctrictive amount and if the calculated cost overpass these amount. If yes we have two solutions: we recalculate the cost accepting an low increased capacity or we consider that the capacity of the arc can not be increased and mark them.

Will obtain a list of costs CA_{ij} for every saturated arc, list that will be sorted ascending.

As we see we consider the value of Δ_{ij} determined for the concrete conditions of every arc. This value is not necessary in every case because a lees value can determined the same results but is more easy to consider this value and eliminate the etap for determining the exact necessary value for increasing the capacity of one arc.

Beginning from the costs list we start the selections of the arcs that can be part of the final solution.

For selecting an arc we consider the minimum cost of increasing operations and the positions of the arc in the network.

Experimentaly we observe that the first arcs that need to be increased are the arcs who determine the cut of network. It is a logical observations and I complete this with another observation. If it is possible it's recommended to increase first the arcs near to the destination nodes.

The condition for finishing the selection process is to obtain an increased transportation network capacity that satisfy the needs in every destination points.

The algorithm is presented to the next:

1. establish the multitude of destination points DP, that need to increase the value of flux and the values of flux for these nodes;

2. establish the multitude of arc candidates (the list of saturated arcs, obtained with Ford-Fulkerson algorithm and verified for additional requests and/or restrictions);

3. calculate the value Δ_{ij} for every arc from the list;

4. calculate the value CA_{ij} for every arc and create the list with these values;

5. sort ascending the list of CA_{ij} costs;

6. **while** (DP $\neq \emptyset$)
   { *Node* ← *Select_nodet*(DN)
   S ← Ø { *S* is the solution set of arcs}
   while not *solution*(*S*) **and** (CA $\neq \emptyset$)
   { *S* ← *select_element*(CA)
   CA ← CA – a_{ij} (selected element of the list)
   S ← *S* ∪ {a_{ij}}
   }
   if *solution(Node,S)* **then write** *S*
   else write "For node "+*Node*+" do not exist solution"
   DN ← DN - *Node*
   }

The function *Select_node*(DN) select an destination node from the multitude of the destination nodes that need an increased value for flux (DN) begininig with the node with maximum value requested.

The function *select_element*(CA) select an arc from the costs list respecting the next rules:

- the selected arc have the minimum cost for increasing operation;

- if exists arcs from the cut of the network then select the arc from the cut of network with minimum cost;

- if not exists arcs from the cut of the network then select the arc with the minimum cost and near to the destination node.

The function *select_element*(CA) need to verify if the increasing of the selected arc can determine an increasing of the network capacity for specified destination node.

Some saturated arc can not determine the increase of the network capacity because they are not in the necessary position. The increase of this arcs capacity can determine only a redistribution of the flux in the network.

For verifying the selected arc, the function *select_element*(CA) can use one of the known methods: dinamic programming, Ford-Fulkerson, preflux method, Edmonds-Karp etc.

The function *solution(Node,S)* evaluates the increased capacity of the network in the specified node and compares them with the value of the necessary network capacity in the same node.

If the capacity generated by increased arcs from *S* is great or equal with the new capacity need for the network in the specified node the process will stopped. Other will continue with the next arc from the list. Maximal value of network capacity could be determined with Ford-Fulkerson method.

The algorithm iterate for a specified until will be determined a solution or the CA lists will be vide.

The process continue to iterate for other destination nodes until the DN will be vide.

The list of candidat arcs will be extracted from results of Ford-Fulkerson algorithm and corrected by eliminating the arcs that could not be increased.

Beginning with this candidate arcs we are sure that we can obtain a minimum cost but not necessary the most minimum possible.

If the problem don't have complete solution after the first application of the algorithm we can reuse them because after we process the network we have a new configuration of the arcs capacity and it is possible to obtain a solution very fast.

4. Conclusion

In the previous paragraph we present an heuristic algorithm for determining the minimal cost of investments for increasing the transportation capacity for a set of destination nodes from a complex transport network.

Because we use at every step the arc with minimum cost for increasing capacity operations we have the premise to obtain the minimum cost of the entire operations.

The algorithm consider the different requests of the destination nodes and also consider the technical and/or economical restrictions forced to the selected arcs

Obviously, the algorithm can not assure the obtaining of the optimal result in every case. Could be situations when the optimal solution need modifications of the network topology or need to extend and the capacity of other arcs, not only the saturated arcs.

Other disadvantage of the method is represented by the necessity to use repeatedly the Ford-Fulkerson algorithm. This algorithm needs a large number of operations in case of complex network and that can affect the eficiency of the proposed algorithm. One ideea to improve the efficiency of the algorithm is to determine at every step the list of the saturated arcs and use the actually list for selecting the arc for the current step. But this solution can generate a very complex algorithms.

5. References

[1] Dorel Dusmanescu, Optimal increase for the transportation network capacity, *WSEAS Transactions on Business and Economics,* Issue 6, Vol. 3, June 2006, p. 473

[2] Dusmanescu Dorel, Stoica Maria, Algoritmi pentru optimizarea retelelor de transport, *Buletinul Universitatii Petrol-Gaze din Ploiesti*, Vol.LVII, No.1, 2005, p. 65-68.

[3] Giuseppe Buttazzo, Aldo Pratelli, Eugene Stepanov, Optimal pricing policies for public transportation networks, *SIAM Journal*, No. 3, 2006, p. 826-853.

[4] Jan Husdal, Reliabilty and Vulnerability versus Costs and Benefits, *Proceedings of the 2nd International Symposiym on Transportation Network Reliability*, Queenstown and Christcurch, New Zealand, 20-24 August, 2004, p. 180-186.

[5] Alesio Bracolini, Giuseppe Buttazzo, Optimal networks for mass transportation problem, *ESAIM Control Optim. Calc. Var.*, No. 1, 2005, p. 88-101.

[6] Zhang Lei, Levinson David Matthew, Economics of Transportation Network Growth, *Transportation Research Board Annual Meeting,* 2006, paper #06-1710.

[7] Ciobanu Gh., Vasile Nica**,** *Cercetari operationale. Optimizari in retele. Teorie si aplicatii economice*, Editura MatrixRom, 2004

[8] Stancioiu Ion, *Cercetari operationale pentru optimizarea deciziilor economice*, Editura Economica, 2004

[9] Thomas Cormen, Charles E. Leiserson, Ronald R. Rivest, *Introducere in algoritmi*, Computer Libris Agora, 2003

[10] www.math.md/imi-site/ro/prog_co.shtml

[11] www.topcoder.com/tc?module=Static&d1=tutorials&d2=maxFlow

[12] www.lix.polytechnique.fr/~durr/MaxFlow

[13] www-cse.uta.edu/~holder/courses/cse2320/lectures/l18/node10.html

[14] www.public.iastate.edu/~roettger/314/projects/ff.html

[15] www.maths.cam.ac.uk/ppa/2005/MPhil/Paper40.ps

[16] www.cs.umass.edu/~barring/cs611/lecture/11.pdf