

n. 475 December 2012

ISSN: 0870-8541

Solving Concave Network Flow Problems

Marta S.R. Monteiro^{1,2}
Dalila B.M.M. Fontes^{1,2}
Fernando A.C.C. Fontes^{3,4}

¹ FEP-UP, School of Economics and Management, University of Porto

² LIAAD/INESC TEC

³ FEUP-UP, Faculty of Engineering, University of Porto

⁴ ISR-Porto

Solving Concave Network Flow Problems *

Marta S.R. Monteiro^{1†}, Dalila B.M.M. Fontes¹, Fernando A.C.C. Fontes²

¹ Faculdade de Economia and LIAAD-INESC TEC

Universidade do Porto

Rua Dr. Roberto Frias, 4200-464 Porto, Portugal.

E-mail: martam@fep.up.pt;fontes@fep.up.pt

Tel.: +351-22-0426240

² Faculdade de Engenharia da Universidade do Porto and ISR-Porto

Rua Dr. Roberto Frias, 4200-464 Porto, Portugal.

E-mail: faf@fe.up.pt

Abstract

The Minimum Cost Network Flow Problem (MCNFP) includes a wide range of combinatorial optimization problems. Many applications exist for MCNFPs for instance supply chains, logistics, production planning, communications and transportations. Concave costs are, in many applications, more realistic than linear ones because of the association of prices with economies of scale. When concave costs are introduced in MCNFPs, then the difficulty to solve them increases and they become NP-Hard. Solution methods developed for these problems comprise both exact and approximate algorithms, the latter ones usually of a heuristic type. What we propose to do in this work is to present an overview of the past and most recent literature published on the subject.

Keywords: Minimum Cost Network Flow Problems, Survey, Heuristics, Exact Methods.

JEL Classifications: C61, C44.

*This work is funded by the ERDF through the Programme COMPETE and by the Portuguese Government through FCT - Foundation for Science and Technology, projects PTDC/EGE-GES/099741/2008 and PTDC/EEA-CRO/116014/2009.

[†]Corresponding author.

1 Introduction

A Minimum Cost Network Flow Problem (MCNFP) can be described as the problem of minimizing the total cost incurred with the distribution of some commodity from the sources to the demand nodes. MCNFPs have a major role in optimization since they include problems such as Transportation Problems (TPs), Assignment Problems (APs), and Shortest Path Problems (SPPs). Therefore, MCNFPs have many practical applications for example in supply chains, logistics, transportation, and facilities location, just to mention but a few (Geunes and Pardalos, 2005).

The costs incurred can take several forms but the ones we are interested in are concave costs, usually associated to economies of scale, discounts, and start-up costs (Guisewite and Pardalos, 1990), which are much more realistic than the linear ones often found in literature and that are considered easy to solve as they are solvable in polynomial time. An example of a situation where concave cost functions have to be accounted for includes the setting of networks of facilities, such as a network of bank branches, that besides the initial costs incurred with the opening of facilities and equipment have also to include operating costs, see (Monteiro and Fontes, 2006). The minimization of a concave function over a convex feasible region, defined by the linear constraints of the problem, makes it much more difficult to solve, therefore more appealing.

Another attractive characteristic of concave MCNFPs is that any Network Flow Problem (NFP) with general nonlinear costs can be transformed into a concave NFP in an expanded network (Lamar, 1993).

In this work, we concentrate our attention in the study of the special case of Minimum concave Cost Network Flow Problems (concave MCNFP). Our objective is to present a review on some methodologies that have been used in order to address MCNFPs.

We start by presenting a formal description of the MCNFP along with its mathematical formulation. We also give a brief characterization of a solution for the concave MCNFP and discuss some issues that define its complexity. An overview of the methodologies used to address this problem is provided next and it is divided accordingly to the three types of concave cost functions considered. We review both exact and heuristic methods. Finally, we close this paper with the conclusions.

2 Concave Minimum Cost Network Flow Problems

A Minimum Cost Network Flow Problem with a general concave cost function can be formally defined as follows.

Given a directed graph $G = (N, A)$, where N is a set of n nodes and A is a set of m available arcs (i, j) , with $i \in N$ and $j \in N$, a concave Minimum Cost Network Flow Problem is a problem that minimizes the total concave costs g_{ij} incurred with the network while satisfying the nodes demand d_j .

Considering the notation summarized bellow,

- n - number of nodes in the network
- m - number of available arcs $(i, j) \in A$
- d_j - demand of node $j \in N$
- x_{ij} - flow on arc $(i, j) \in A$
- y_{ij} = binary variable assuming the value 1 if arc $(i, j) \in A$ is chosen and 0 otherwise
- g_{ij} - concave cost function of arc $(i, j) \in A$
- u_{ij} - upper limit on flow through arc $(i, j) \in A$
- l_{ij} - lower limit on flow through arc $(i, j) \in A$

the mathematical model for the concave MCNFP can then be written as follows:

Model 1 A mixed-integer mathematical programming model for the general concave MCNFP problem.

$$\min: \sum_{(i,j) \in A} g_{ij}(x_{ij}, y_{ij}) \quad (1)$$

$$\text{s.t.}: \sum_{\{i|(i,j) \in A\}} x_{ij} - \sum_{\{k|(j,k) \in A\}} x_{jk} = d_j, \forall j \in N, \quad (2)$$

$$0 \leq x_{ij} \leq u_{ij}, \quad \forall (i,j) \in A, \quad (3)$$

$$x_{ij} \geq l_{ij}y_{ij}, \quad \forall (i,j) \in A. \quad (4)$$

The objective is to minimize the total costs defined in (1), provided that the demand is satisfied, stated by the *flow conservation constraints* (2), and that the arcs *capacity constraints* in (3) and (4) are not violated. Regarding the demand, d_j takes a negative or positive value depending on whether j is a source or a demand node, respectively. We assume that the total source demand equals the total sink demand, thus $\sum_{j \in N} d_j = 0$. Sometimes neither upper nor lower bounds are established for the flows in the arcs, therefore the problem is considered uncapacitated which mathematically translates into $u_{ij} = +\infty$ and $l_{ij} = 0$.

Regarding concave cost functions, they can take several forms but the most popular ones used in literature are $b_{ij} \cdot x_{ij} + c_{ij}$, also known as concave fixed-charge functions. However, we can also find other concave cost functions such as the square root cost function $b_{ij} \cdot \sqrt{x_{ij}}$ that has been considered by Altiparmak and Karaoglan (2006), and the second-order polynomial cost

function $-a_{ij} \cdot x_{ij}^2 + b_{ij} \cdot x_{ij}$ used by Dang et al (2011), just to mention but a few possibilities.

2.1 Characterization of a solution for the concave MCNFP

A feasible solution for the concave MCNFP is a solution that does not violate neither (2) nor (3). Lozovanu (1983) observed that if a feasible solution exists for concave MCNFPs, then an optimal solution must occur at a vertex, i.e. an extreme point, of the convex polyhedron defined by the problem constraints (2) and (3). Also, the minimization of a concave cost function in a convex polyhedron means that a local optimum does not imply a global optimum. Thus, in order to find the global optimum solution for this problem, the set of all extreme points in the convex polyhedron has to be searched for.

Furthermore, if the function has a finite global minimum on the feasible region, then there is an extreme solution that is an optimal solution (Eggleston, 1963).

2.2 Complexity

The complexity of an optimization problem is a very important issue mainly because it will allow the researcher to choose an adequate method to solve it. For example, if the MCNFP instance to be solved is considered easy, an exact method, such as simplex or branch-and-bound, can be used, whereas if it is considered hard then a heuristic method is probably more adequate as it can provide a fairly good solution in a small amount of time. In this section, we provide an insight on the main characteristics of MCNFPs that are associated with the degree of their complexity.

The cost function considered in an optimization problem can have a great impact on the difficulty to solve it. We have already mentioned that MCNFPs with linear costs are considered easy to solve. However, if concave costs are used the difficulty to solve them increases and they become NP-Hard (Guisewite and Pardalos, 1991a). The complexity arises from the fact that in the minimization of a concave function (even over a convex feasible region) a local optimum is not necessarily a global optimum. Guisewite and Pardalos (1991b) provide a study on how the form of the concavity affects the complexity of these problems. The authors use functions with the following form $\alpha_{ij}x_{ij}^{\beta_{ij}}$. They were able to provide evidence that, on the one hand the number of local optima increases with the decrease of β_{ij} , and, on the other hand the larger the set from which to draw the value of α_{ij} , the smaller the set of local optima. Problems with fixed-charge costs are a special case of concave optimization and they may be simpler or harder to solve accordingly to characteristics that have been argued by Kennington and Unger (1976), Palekar et al (1990), and Barr et al (1981). One such characteristic is the ratio between fixed (F) and variable (C) costs $\frac{F}{C}$. On the one hand, Kennington and Unger (1976) claim that the difficulty to solve fixed-charge problems increases with this ratio. On the other hand, Palekar

et al (1990), which disagree with them, suggest that only ratios with intermediate values are difficult to solve because if the ratio is very small or very large the problem is easier to solve either because fixed costs are negligible thus transforming the problem into a linear one, or because the problem reduces to the one of minimizing fixed costs. The special case of the Single Source Uncapacitated MCNFP with fixed-charge costs has been proven to be NP-Hard (Hochbaum and Segev, 1989).

Another issue usually related to the complexity of a MCNFP is the density of the network to be considered, that is the ratio between the m available and all existing arcs in a network. It is easy to conclude that the denser the network the harder the problem is to solve, because the number of feasible solutions increases and so does the computational time needed to enumerate all of them in case of an exact method.

The number of arcs with nonlinear cost is also a major factor affecting the difficulty to solve a nonconvex MCNFP (Tuy et al, 1995). The larger the number of nonlinear arcs, the harder the problem becomes. Some problems with a small number of arcs with nonlinear costs have been proven to be solvable in polynomial time, e.g. (Guisewite and Pardalos, 1993).

Regarding the capacity of arcs in a network, both versions of the concave MCNFP, Capacitated and Uncapacitated, are known to be NP-Hard.

In network flow problems, demand nodes are usually the ones contributing to the complexity of a problem because transshipment nodes represent a null cost bridge between demand nodes. In addition, problems with several source nodes can be transformed into problems with a single source node (Zangwill, 1968). Therefore, the size of a problem, and consequently one of the many aspects contributing to the difficulty in solving it, is usually related with the number of demand vertices.

3 Solution Methods for MCNFPs

Most of the works developed around concave MCNFPs consider fixed-charge costs, that is cost functions having a fixed start-up component and a linear routing component. Other works considering nonlinear concave routing costs (Guisewite and Pardalos, 1991a; Horst and Thoai, 1998; Smith and Walters, 2000) do not include a fixed component.

As far as we are aware of, only a few works consider concave cost functions made of nonlinear concave routing costs and fixed costs simultaneously, which are those of Burkard et al (2001), Fontes et al (2003, 2006b,a), Fontes and Gonçalves (2007), and Dang et al (2011). This is the main reason why the review of previous works is mainly on the Fixed-Charge problem.

Exact solution methods are usually not very efficient in the case of NP-hard problems, because they make use of implicit or explicit enumeration of the vertices (local optima) of the convex

polyhedron defined by the flow conservation constraints. Nonetheless, exact methods are very important in the sense that they provide us with optimal values, even if it is only for small problem instances, and due to the theoretical advances they unravel.

The most popular methods to solve NP-Hard problems are heuristic methods. Low usage of computational memory and computational time are their most attractive characteristics although they may provide only a local optimum. Heuristic methods may be classified, regarding the number of solutions evaluated, into single-point or multi-point algorithms. Generally speaking, a single-point algorithm evaluates a single solution in each phase of the search. These algorithms are usually coupled with a local search procedure in order to improve the solution. Examples of such heuristics are Simulated Annealing and Tabu Search. Multi-point heuristics, in opposition, analyse a set of solutions in each phase/iteration and usually combine the best parts of the solutions in order to create new solutions. Examples of these are Evolutionary Algorithms, such as Genetic Algorithms, and Ant based algorithms. Furthermore, hybrid algorithms are also very popular because they usually join forces between methods focused in the exploration of the search space and methods, such as local search, more focused in the exploitation of the search space.

This section is divided into three parts accordingly to the type of concave cost function considered: nonlinear routing costs with and without a fixed component, and linear routing costs with a fixed charge component.

3.1 Nonlinear concave routing costs with fixed charge components

Burkard et al (2001) develop a Dynamic Programming algorithm, to solve the SSU concave MCNFP, based on linear approximations of the cost function, where concave costs are given by $c + bx_{ij} + ax_{ij}^d$, with $d \in [0, 1]$ and where a, b, c , and d might or might not depend on the arc (i, j) . The authors develop a DP algorithm to solve it and prove that with the use of approximated linear cost functions the method converges towards an optimal solution. The method is only adequate to networks where nodes have small degrees. Therefore, although they are able to solve problems with 1103 nodes they may only have up to 2203 arcs.

Upper Bounds (UBs) based on local search are calculated by Fontes et al (2003) to solve SSU concave MCNFPs. The local search is based on swaps of arcs and is performed repeatedly with different initial solutions, this way avoiding getting trapped into a local optimum. Given an initial feasible solution, and for every subtree T_y in the solution, the Local Search procedure tries to put T_y “under” another node k that does not belong to that subtree. If a new solution, thus constructed, has a better cost, the UB is updated and the procedure continues to the next subtree. When no more reductions in the cost can be found the algorithm stops. The initial feasible solution is provided by a Lower Bound (LB), found by a relaxation of the state space of a DP recursion (Fontes et al, 2006c), and it consists of a network supplying a set of demand

nodes. Supplied nodes are added to the set of fixed-nodes and the rest are added to a temporary nodes list. Then, the temporary nodes are appended to the solution tree. Each temporary node k is selected, one at the time, and the arc linking it to the LB tree is identified as the one representing the lowest cost for the path linking the source node and node k . This action is performed until the set of temporary nodes is empty, and a new improved solution is found.

Another BB procedure is proposed by Fontes et al (2006a) to optimally solve SSU concave MCNFPs considering fixed-charge and nonlinear concave second-order complete polynomial cost functions. At each node of the BB procedure, a lower bound for the cost of the solution is found by making use of a modified relaxation of the state space of the DP developed by Fontes et al (2006b). The relaxation only guarantees that the number of used arcs is the correct, i.e. $n - 1$ where n is the number of nodes. However, any arc may be used several times. The BB procedure is as follows. Given an LB solution, a branching arc (i, j) is chosen, and two branches are identified and analysed, one where the arc is deleted from the solution and the other where it is forced to be in the solution. If, when (i, j) is deleted from the solution, any demand node is disconnected from the solution tree, then that branch is discarded, otherwise lower and upper bounds are obtained. After analysing that branch, the algorithm steps into the other branch, where (i, j) is fixed as part of the solution and again upper and lower bounds are calculated. The choice of the BB tree node to go to next is made by selecting the node with the largest gap between the corresponding LB and the best upper bound available. An upper bound is computed as explained above and given by Fontes et al (2003).

Fontes et al (2006b) use an exact method involving DP to optimally solve SSU concave MCNFPs with four cost functions: linear, fixed-charge, and second-order polynomials both with and without a fixed-charge component. The state space graph is gradually expanded by using a procedure working in a backward-forward manner on the state space graph. The dynamic part of the algorithm is related to the identification of only the states needed for each problem being solved. The DP procedure has as many stages as the number of nodes $n + 1$ in the problem to be solved, and each stage is made up a set of states $s_i \equiv (S, x)$ such that each state considers a subset S (of the set of nodes W) to be supplied and some root node x , with $x \in S$. Therefore a stage is given by the cardinality of S . The algorithm starts from the final state where all demand nodes are considered along with the root node t , (W, t) . Then, it moves backwards, until some state already computed is reached, identifying possible states in the way. Then, it moves forward, through already computed states, until a not yet computed state (S, x) is reached. The algorithm continues this backward-forward procedure until the last stage (W, t) is reached and no more moves can improve its cost, and thus the optimal solution has been found.

Lower bounds for SSU concave MCNFPs, derived from state space relaxations, are given by Fontes et al (2006c). The State Space Relaxation associated with a DP recursion can be translated into a reduction on the number of states, by forcing constraints in the linear programming formulation to appear as variables of the DP. The authors provide a new relaxation adding a

new constraint to the q-set relaxation forcing the solution, of a problem with $n + 1$ arcs, to have exactly n nodes. The solution is a LB since the n arcs used are not necessarily all different. The bound obtained is further improved by penalizing demand nodes not fully supplied. These LBs are later on used in the bounding phase of a Branch-and-Bound procedure given by Fontes et al (2006a).

Kim et al (2006) introduced Tabu Search strategies into the basic DSS having improved upon the results of the basic DSS developed by Kim and Pardalos (1999). The new algorithm is called Enhanced DSS and has three phases. The first phase runs the basic DSS with an addition, whenever the best solution to the moment is updated, the arcs with the largest changes on the flow, when compared to the previous iteration, are added to a set called the inspiration set ξ . Also, a record of the frequency of appearance of each arc with a positive flow is incremented. After reaching one of the DSS stopping criteria, the intensification phase follows. In it, some arcs are chosen to be tabu, according to the frequency of their appearance in previous solutions. Other arcs, including the ones in ξ , are added to a candidate arcs list α which will be the ones allowed to enter new solutions. Once these sets are identified, the DSS phase is run again. The initial linearisation factors used for those arcs not in α , in the DSS phase that follows the intensification phase, are the same linear factors associated with the arcs of the most recently improved best solution. At the end of the intensification phase, the third phase, the diversification phase takes place in order to explore new regions of the search space. Arcs appearing not so frequently are added to the candidate list β , based on information about the reduced costs, i.e. the amount by which \bar{c}_{ij} has to be improved in order for arc (i, j) to enter the solution. Tabu and non-tabu lists are also maintained during this phase. The DSS phase is run again but now using the reduced costs as the initial linearisation factors. Both the intensification and the diversification are run a fixed number of times. The tabu mechanisms introduced in this DSS were inspired by the TS heuristic previously developed by Sun et al (1998) to solve Fixed-Charge Transportation Problems which, to the moment, and along with the one of Glover et al (2005), and more recently of Aguado (2009), is still one of the most efficient heuristic methods to solve such an NP-hard problem.

Fontes and Gonçalves (2007) use a genetic algorithm coupled with a local search procedure, which was called HGA, to solve the Single-Source Uncapacitated Minimum Cost Network Flow Problem (SSU MCNFP) with general concave costs. Random keys are used to encode the chromosome, as they allow all solutions generated by crossover to be feasible solutions. In order to create a new generation of solutions, the algorithm selects the top chromosomes, regarding their fitness value, which are directly copied onto the next generation. The mutation operator used, not a traditional one, generates new random chromosomes, without any genetic influence on the current population. Finally, the remaining chromosomes, to integrate the next generation, are created by applying a biased probability crossover operator. The crossover between two parent solutions is performed by considering a gene at the time. The algorithm generates a

vector with as many random numbers (in the interval $[0, 1]$) as the genes in a chromosome. Every random number on that vector is tested and if its value is lower than a certain probability, say 70%, then the gene of the offspring is drawn from the best parent, otherwise it is drawn from the other parent. This way, better parents pass on more genetic information. The local search procedure, which is applied to all solutions, consists of swap operations between arcs already in the solution and arcs not in the solution. Arcs (i, j) belonging to the solution tree are sorted and considered in descending order of nodes priority. Then each arc (k, j) outside the solution tree, is considered in descending order of priority, and the first one that does not introduce a cycle in the solution is the one chosen to substitute the leaving arc (i, j) . When compared with results in literature, the HGA was able to improve upon upper bounds provided by a heuristic algorithm based on local search, as well as running times.

Poorzahedy and Rouhani (2007) solve Transportation Network Design problems by proposing seven hybrid algorithms based on a previously developed Ant System (Poorzahedy and Abulghasemi, 2005) and on three improvements with notions borrowed from genetic algorithms, simulated annealing and tabu search. The first improvement introduced modifies the way pheromones are updated, allowing only the three best solutions to contribute to the pheromone update. The second improvement is based on evolutionary algorithms and it allows mutation to take place under some conditions. The mutation is applied in substitution of the construction phase, and it occurs in the middle of the run of the algorithm, that is to say, in the fifth iteration since the algorithms are allowed to run only 10 iterations. The 3 best solutions of each of the previous four iterations, are retained. These solutions will be used to calculate the frequency of appearance of each project. Then, the 2 best solutions of the previous iterations are also retained along with the 2 best solutions of them all, with repeated solutions allowed. Repeated solutions identify the least, or next least, frequent project and substitute it with the most, or next most, frequent project provided that the solution is still feasible. The solutions thus found are considered new solutions and the algorithm continues to the next step, the pheromone update. The last improvement, applied from the second iteration onwards, is based on Simulated Annealing concepts and its purpose is to reduce the computational effort of computing net benefits by decreasing the probability of solutions with low levels of energy, as opposite to the usual simulated annealing. The seven hybrid algorithms are constructed by incorporating into the AS different combinations of these three improvements, as well as incorporating each one on its own. The algorithms were applied to a real-size traffic-network of a city in Iran and the algorithm incorporating all three improvements achieved the best results of them all.

More recently, Monteiro et al (2012) address the SSU MCNFP with concave cost functions by developing an Ant Colony Optimization (ACO) algorithm to solve it. The ACO algorithm is hybridized with a local search procedure (HACO) in order to improve its performance. The cost functions considered are of three types, a fixed-charge function $bx_{ij} + c_{ij}$ and two second order polynomials, one with and another without a fixed charge component, that is $ax_{ij}^2 + bx_{ij} + c_{ij}$

and $ax_{ij}^2 + bx_{ij}$, respectively. Also, all arcs have associated nonlinear and concave costs. The ACO algorithm is based on the min-max ant system (Stützle and Hoos, 1997) in the sense that it uses pheromone bounds to avoid the fast convergence of the pheromone trail. The authors provide a study on the performance of the algorithm with the variation of the parameters values, which revealed that some are of vital importance for the good performance of the algorithm, while others can be set to almost any reasonable value within the problem context. Local search is applied right after all ants have constructed their solution. The algorithm identifies the best solution found by the ants at the current iteration and local search is performed to it and also to other four randomly chosen solutions. The gap results obtained with the ACO algorithm were always as good or better than the ones reported in literature (Fontes and Gonçalves, 2007). Furthermore, the computational time requirements of the ACO algorithm were much lower, even when compared with the ones obtained with CPLEX for large problem instances.

3.2 Nonlinear concave routing costs without a fixed cost component

The most common techniques associated with exact methods to solve MCNFPs are usually Branch-and-Bound (BB) and Dynamic Programming (DP). Both techniques approach the problem by dividing it into smaller subproblems which, in turn, are divided into smaller subproblems, and so on.

The branch-and-bound procedure developed by Soland (1974), is still very popular and used by other authors as a basis for their own branch-and-bound methods. The idea is to use linear underestimation by convex envelopes and to use rectangles defined by the capacity flow constraints to partition the search space. In Soland's algorithm the branching procedure starts by considering the rectangle defined by the capacity flow constraints C . Then, a subset $C^a \subset C$ is partitioned into two subrectangles C^b and C^c such that $C^b \cup C^c = C^a$. This way, a subproblem at a node b has its domain defined by both the rectangle C^b and the flow constraints. The bounding procedure corresponds to the computation of a lower bound on the optimal solution found in the subrectangle C^a . This lower bound is obtained by solving a linear relaxation of subproblem C^a .

Gallo et al (1980) developed a BB algorithm to solve Single Source Uncapacitated concave MCNFPs (SSU MCNFPs). In the problems to be solved the authors consider nonnegative separable concave cost functions for all arcs, however only some of the n nodes are demand nodes, the others being merely transshipment nodes. The BB algorithm initially starts with only the source node and the branching part of it is performed by adding arcs extending the current subtree. Then, lower bounds are obtained for each BB node by using linear underestimation of the arcs costs for demand nodes not satisfied. Latter on, Guisewite and Pardalos (1991a) improve these lower bounds by projecting them on the cost of extending the current path.

Horst and Thoai (1998) consider the capacitated version of concave MCNFPs where a fixed

number of arcs have concave flow costs and the other arcs have linear costs. A BB algorithm based on the work of Soland (1974) is developed leading to improvements of lower bounds. This algorithm differs from Soland's in two ways: in the way rectangles are subdivided, turning them into integral rectangles of approximately the same size, and in the way branching arcs are chosen, in this case from the set of arcs with nonlinear costs. A survey on MCNFPs with a fixed number of arcs with nonlinear costs can be found at (Tuy, 2000).

Genetic algorithms are heuristic algorithms based on the evolution of species and the main idea is to take a set of solutions, which are called a population or generation, and to combine the best of them, following the maxima "*the survival of the fittest*", in order to generate new improved solutions. A mutation factor is also usually incorporated.

Smith and Walters (2000) provide a heuristic method based on Genetic Algorithms to find minimum cost optimal trees on networks and apply it to the solution of SSU concave MCNFPs. The cost functions considered are concave given by the square root of the flow. Randomly generated feasible trees are considered for the initial population. The authors stress out the problematic of generating feasible trees specially in the mutation and the crossing of parents and propose a technique for each. Accordingly to their fitness value, two parents at a time, T_1 and T_2 , are chosen to reproduce thus creating two new trees. In order to accomplish that, a bipartite graph is created by overlapping T_1 and T_2 . The children have a common structure constituted by the parents common arcs. The number of arcs unique to each parent is the same. Therefore, these arcs are chosen in pairs, one from each parent, and one of them is attributed to one child and the other to its sibling, with a probability of 0.5. If at least one child is not a tree the crossing process is repeated until both of them are. The mutation operator is applied to a subset of the population, and is defined so that one arc is randomly chosen to be substituted by another one in such a way as to maintain the solutions feasibility, that is, so that the solution is still a tree.

A hybrid between Simulated Annealing and Tabu Search with an adaptive cooling strategy is the algorithm proposed by Altıparmak and Karaoglan (2006) to solve the Concave Cost Transportation Problem, where the cost function is proportional to the square root of the flow $c_{ij}\sqrt{x_{ij}}$. After the generation of an initial feasible solution, swap moves between an arc in the solution and an arc not in the solution are applied in order to improve the solution. An arc is added to the solution, thus creating a cycle. As in a pivot move on the network flow simplex algorithm, in order to maintain the feasibility of the solution the flow of the arcs in the cycle is increased or decreased, as needed, accordingly to the flow on the arc to be dropped from the solution. The set D of arcs from the cycle whose flow must be adjusted by being decreased is identified and the arc $(k, l) \in D$ with the least amount of flow is the one to be dropped from the solution. The tabu procedure is incorporated in the algorithm in the form of two tabu lists, one keeping track of the arcs leaving the solution and another one keeping track of the arcs entering the solution. This way, the number of arcs to be tested decreases, and consequently the computational

time also decreases. The adaptive cooling schedule is based on a ratio between the temperature at the previous iteration and 1 minus the cubic root of the temperature, allowing for a slower temperature decreasing rate.

Dang et al (2011) developed a deterministic annealing algorithm for the capacitated version of the concave MCNFP, that can be used to solve both single-source and multiple-source cases. They use of a Hopfield type barrier function, which is a notion borrowed from the theory of neural networks, to cope with the lower and upper bounds on the capacities of the arcs. Each arc (i, j) is associated to a Hopfield-type barrier field thus allowing the capacity constraints to be incorporated into the objective function. The barrier parameter has a behaviour similar to the temperature on the simulated annealing, decreasing towards zero, from a large positive number. The linear constraints, the flow conservation constraints, are dealt with the use of Lagrangean Multipliers, by incorporating them into the objective function. This way, a Lagrange and barrier function is obtained. Numerical results are provided, for problems with 5 up to 12 nodes, for both a linear cost function $b_{ij} \cdot x_{ij}$ and a concave second order polynomial function $-a_{ij} \cdot x_{ij}^2 + b_{ij} \cdot x_{ij}$.

3.3 Linear routing costs with a fixed-charge component

Methods based on the linearisation of the cost function are very popular to solve fixed-charge problems.

In (Kennington and Unger, 1976) a linear relaxed version of the Fixed-Charge Transportation problem is used, where the usual fixed-charge objective function is replaced by $d_{ij} \cdot x_{ij}$ with $d_{ij} = c_{ij} + f_{ij}/u_{ij}$, where u_{ij} represents the flow capacity of arc (i, j) . This relaxation is used to obtain bounds for the solution of the original problem, which are later strengthened using Driebeek penalties (Driebeek, 1966), which are used in the branching and fathoming phases of a BB algorithm.

Kim and Pardalos (1999) developed a technique called Dynamic Slope Scaling (DSS) in order to solve the well-known NP-Hard Fixed-charge Network Flow Problem. Given an objective function of the type $f(x) = \sum_{(i,j)} c_{ij}x_{ij} + s_{ij}$, where c_{ij} represents the flow variable cost, and s_{ij} represents the fixed cost, the idea behind it is to find a linear factor that can represent the variable and fixed costs at the same time. Thus iteratively solving linear problems. At each iteration the cost function is updated by using the information of the solution found in the previous iteration. The algorithm follows these two steps, solving the linear problem and updating the cost function, until two consecutive iterations return the same solution. Later on, Kim and Pardalos (2000) extend the use of DSS by incorporating a local search scheme, called Trust Interval, to solve concave piecewise linear NFPs. An adaptation of the DSS technique, coupled with a local search procedure, was also used by Monteiro and Fontes (2006) to solve the problem

of bank-branch location and sizing with fixed-charge costs.

Ortega and Wolsey (2003) solve the Uncapacitated Fixed-Charge Network Flow problem with a Branch-and-Cut algorithm by extending the cutting planes previously used to solve uncapacitated lot sizing problems, and applying them to a commercial optimisation routine of software Xpress. The problem is formulated as a Mixed Integer Problem (MIP) where binary variables y_{ij} , associated to the use of arc (i, j) are considered. Four dicut-inequalities are defined as follows: simple dicut, mixed dicut, simple inflow-outflow, and mixed dicut with outflow inequalities. However, only dicut-inequalities and simple inflow-outflow inequalities are used, due to their performance in preliminary tests. Another feature therein introduced was the use of a dynamic set node list for the dicut inequalities. Single-commodity and multicommodity problems have been solved.

A recent work on MCNFPs is the one of Nahapetyan and Pardalos (2007) where the authors consider a concave piecewise linear cost function. The problem is transformed into a continuous one with a bilinear cost function, through the use of a nonlinear relaxation technique. First, the problem is formulated as a mixed integer program, by introducing the usual binary variables y_{ij}^k associated to the fixed costs, where k identifies the linear segment of the cost function. Then, the binary nature of y_{ij}^k and constraint $x_{ij}^k \leq M y_{ij}^k$ are replaced with $y_{ij}^k \geq 0$ and $x_{ij}^k = x_{ij} y_{ij}^k$, respectively, where x_{ij} is the flow in arc (i, j) . The relaxed problem is then solved with a dynamic slope scaling method, based on the one proposed by Kim and Pardalos (1999, 2000) and explained above. Nahapetyan and Pardalos (2008) improve upon the results of the original DSS (Kim and Pardalos, 1999) by approximating the fixed-charge cost function by a concave piecewise linear function. The problem is transformed into a continuous one with a bilinear cost function. This approach is considered a novelty because fixed-charge functions are usually approximated to linear functions. One of the cost functions represents a line connecting the origin and some point $(\varepsilon_{ij}, f(\varepsilon_{ij}))$, and is defined as $f_{ij}^{\varepsilon_{ij}}(x_{ij}) = c_{ij}^{\varepsilon_{ij}} x_{ij}$ if $x_{ij} \in [0, \varepsilon_{ij}]$. The other one is defined as $f_{ij}^{\varepsilon_{ij}}(x_{ij}) = c_{ij} x_{ij} + s_{ij}$ for $x_{ij} \in [\varepsilon_{ij}, \lambda_{ij}]$, where λ_{ij} is the capacity of arc (i, j) and $c_{ij}^{\varepsilon_{ij}} = c_{ij} + s_{ij}/\varepsilon_{ij}$ with c_{ij} as the flow cost and s_{ij} the fixed cost. Although the arcs are capacitated, this problem can be transformed into an uncapacitated one by substituting the capacities with a sufficiently large M (constant). The problem thus formulated, a value for $\varepsilon_{ij} \in [0, \lambda_{ij}]$ is chosen and the problem is then solved by the DSS algorithm developed by Nahapetyan and Pardalos (2007). At the end of each iteration, every flow variable x_{ij} is tested in order to verify if its value is within $[0, \varepsilon_{ij}]$. If so, ε_{ij} is decreased by a constant $\alpha \in]0, 1[$, such that $\varepsilon_{ij} \leftarrow \alpha \varepsilon_{ij}$, otherwise the algorithm stops and the best found solution is returned.

Rebennack et al (2009) propose a continuous bilinear formulation from which an exact algorithm, based on the algorithm developed earlier by Nahapetyan and Pardalos (2007, 2008), is derived to solve fixed-charge MCNFPs. The fixed-charge function is modified by introducing binary variables y_{ij} , defined for all arcs, that take the value 1 if x_{ij} , the flow on arc (i, j) , is between a given small value ε_{ij} and the capacity of arc (i, j) , and 0 if x_{ij} is between 0 and ε_{ij} .

The relaxation of these variables results on a continuous bilinear network flow problem with the following cost function

$$f(x, y) = \sum_{(i,j) \in A} \left(c_{ij}^{\epsilon} x_{ij} + \left(s_{ij} - \frac{s_{ij}}{c_{ij}} x_{ij} \right) y_{ij} \right), \quad (5)$$

where A is the set of all available arcs, c_{ij} is the variable cost of arc (i, j) , and s_{ij} is the fixed cost of arc (i, j) . The algorithm defined for this new formulation proved to converge in a finite number of steps.

Another work on Fixed-Charge (FC) problems is the one of Adlakha et al (2010), where the authors make use of the relaxation of the binary restriction on the y_{ij} value, which was initially proposed by Balinski (1961). The optimal solution of this relaxation, has the property that the value per unit flow in each arc becomes

$$C_{ij} = c_{ij} + \frac{f_{ij}}{\min(s_i, d_i)}. \quad (6)$$

The relaxed problem becomes a linear one. The objective function value of the optimum solution for this new problem provides the FC with a lower bound to the total flow costs, while the objective function value for the FC provides an upper bound. Then, based on the differential of the fixed costs for the FC and for the relaxed problem, the algorithm iteratively chooses demand nodes to be provided with all their demand by a single supply, adjusting the rest of the network by eliminating the most expensive arc. The bounds are then tightened until an optimum value is reached and both bounds have the same value. Although the authors provide a numerical example to illustrate the branching procedure, they do not provide computational results.

4 Conclusion

Concave cost functions are usually associated to economies of scale, thus they are very interesting and important from the point of view of logistics, transportation, and supply-chains, just to mention but a few areas. Nevertheless, although they are, usually, present in practical applications, surprisingly not much has been done in academic studies regarding network flow problems with cost functions. In fact, in recent years there even has been a slow down in research for this class of problems, when compared to other classes. The main reason for this to happen may have to do with the complexity concave cost functions bring to the solvability of the problem. In this work, we have mainly reviewed works on Nonlinear concave Minimum Cost Network Flow Problems. In recent years some of these problems have been solved with heuristic methods that, although not guaranteeing a global optimal solution, are usually able to find a good solution rapidly, perhaps a local optimum. There is still much to improve regarding

the results that have been obtained, either because exact methods cannot cope with the size of the problem or because solutions found by heuristics can be further improved. With the recent boom of nature based heuristic algorithms, such as, for example, the Bees Algorithm (Pham et al, 2005) or the Water Drops Algorithm (Shah-Hosseini, 2009), it is expectable to have some of the existing results improved.

References

- Adlakha V, Kowalski K, Lev B (2010) A branching method for the fixed charge transportation problem. *Omega* 38:393–397
- Aguado JS (2009) Fixed charge transportation problems: a new heuristic approach based on lagrangean relaxation and the solving of core problems. *Annals of Operations Research* 172:45–69
- Altıparmak F, Karaoglan I (2006) An adaptive tabu-simulated annealing for concave cost transportation problems. *Journal of the Operational Research Society* 59(3):1–11
- Balinski ML (1961) Fixed cost transportation problem. *Naval Research Logistics Quarterly* 8:41–54
- Barr F, Glover F, Klingman D (1981) A new optimization method for fixed charge transportation problems. *Operations Research* 29:448–463
- Burkard RE, Dollani H, Thach PT (2001) Linear approximations in a dynamic programming approach for the uncapacitated single-source minimum concave cost network flow problem in acyclic networks. *Journal of Global Optimization* 19:121–139
- Dang C, Sun Y, Wang Y, Yang Y (2011) A deterministic annealing algorithm for the minimum concave cost network flow problem. *Neural Networks* 24:699–708
- Driebeek NJ (1966) An algorithm for the solution of mixed integer programming problem. *Management Science* 12:576–587
- Eggleston HG (1963) Convexity. *Cambridge tracts in mathematics and mathematical physics*
- Fontes DBMM, Gonçalves JF (2007) Heuristic solutions for general concave minimum cost network flow problems. *Networks* 50:67–76
- Fontes DBMM, Hadjiconstantinou E, Christofides N (2003) Upper bounds for single-source uncapacitated concave minimum-cost network flow problems. *Networks* 41(4):221–228
- Fontes DBMM, Hadjiconstantinou E, Christofides N (2006a) A branch-and-bound algorithm for concave network flow problems. *Journal of Global Optimization* 34:127–155

- Fontes DBMM, Hadjiconstantinou E, Christofides N (2006b) A dynamic programming approach for single-source uncapacitated concave minimum cost network flow problems. *European Journal of Operational Research* 174:1205–1219
- Fontes DBMM, Hadjiconstantinou E, Christofides N (2006c) Lower bounds from state space relaxations for network routing problems. *J Global Optim* 34:97–125
- Gallo G, Sandi C, Sodini C (1980) An algorithm for the min concave cost flow problem. *Euro* 4:249–255
- Geunes J, Pardalos P (2005) *Supply Chain Optimization*. Springer, Berlin
- Glover F, Amini M, Kochenberger G (2005) Parametric ghost image processes for fixed-charge problems: a study of transportation networks. *Journal of Heuristics* 11:307–336
- Guisewite GM, Pardalos PM (1990) Minimum concave-cost network flow problems: Applications, complexity, and algorithms. *Annals of Operations Research* 25:75–99
- Guisewite GM, Pardalos PM (1991a) Algorithms for the single-source uncapacitated minimum concave-cost network flow problem. *Journal of Global Optimization* 3:245–265
- Guisewite GM, Pardalos PM (1991b) Global search algorithms for minimum concave-cost network flow problems. *Journal of Global Optimization* 1:309–330
- Guisewite GM, Pardalos PM (1993) A polynomial time solvable concave network flow problem. *Networks* 23:143–147
- Hochbaum DS, Segev A (1989) Analysis of a flow problem with fixed charges. *Networks* 19:291–312
- Horst R, Thoai NV (1998) An integer concave minimization approach for the minimum concave cost capacitated flow problem on networks. *OR Spectrum* 20:47–53
- Kennington J, Unger V (1976) A new branch-and-bound algorithm for the fixed charge transportation problem. *Management Science* 22:1116–1126
- Kim D, Pardalos P (2000) Dynamic slope scaling and trust interval techniques for solving concave piecewise linear network flow problems. *Networks* 35(3):216–222
- Kim D, Pardalos PM (1999) A solution approach to the fixed charge network flow problem using a dynamic slope scaling procedure. *Networks* 35:216–222
- Kim D, Pan X, Pardalos PM (2006) An enhanced dynamic slope scaling procedure with tabu scheme for fixed charge network flow problems. *Computational Economics* 27:273–293

- Lamar BW (1993) *Network Optimization Problems: Algorithms Applications and Complexity*, World Scientific, chap A method for solving network flow problems with general nonlinear arc costs, pp 147–167
- Lozovanu D (1983) Properties of optimal solutions of a grid transport problem with concave function of the flows on the arcs. *Engineering Cybernetics* 20:34–38
- Monteiro MSR, Fontes DBMM (2006) Locating and sizing bank-branches by opening, closing or maintaining facilities. In *Proceedings of the International Scientific Annual Conference Operations Research 2005 - OR2005*, VerlagMonteiro2006
- Monteiro MSR, Fontes DBMM, Fontes FACC (2012) Concave minimum cost network flow problems solved with a colony of ants. *Journal of Heuristics* (To appear)
- Nahapetyan A, Pardalos P (2007) A bilinear relaxation based algorithm for concave piecewise linear network flow problems. *Journal of Industrial and Management Optimization* 3:71–85
- Nahapetyan A, Pardalos P (2008) Adaptive dynamic cost updating procedure for solving fixed charge network flow problems. *Computational Optimization and Applications* 39:37–50
- Ortega F, Wolsey LA (2003) A branch-and-cut algorithm for the single-commodity, uncapacitated, fixed-charge network flow problem. *Networks* 41:143–158
- Palekar US, Karwan MH, Zionts S (1990) A branch-and-bound method for the fixed charge transportation problem. *Management Science* 36(9):1092–1105
- Pham D, Ghanbarzadeh A, Koç E, Otri S, Rahim S, Zaidi M (2005) The bees algorithm - a novel tool for complex optimisation problems. Tech. rep., Manufacturing Engineering Centre, Cardiff University, UK
- Poorzahedy H, Abulghasemi F (2005) Application of ant system to network design problem. *Transportation* 32:251–273
- Poorzahedy H, Rouhani OM (2007) Hybrid meta-heuristic algorithms for solving network design problem. *European Journal of Operational Research* 182:578–596
- Rebennack S, Nahapetyan A, Pardalos P (2009) Bilinear modeling solution approach for fixed charge network flow problems. *Optimization Letters* 3:347–355
- Shah-Hosseini H (2009) The intelligent water drops algorithm: a nature-inspired swarm-based optimization algorithm. *International Journal of Bio-Inspired Computation* 1:71–79
- Smith DK, Walters GA (2000) An evolutionary approach for finding optimal trees in undirected networks. *European Journal of Operational Research* 120(3):593 – 602, DOI DOI: 10.1016/S0377-2217(98)00385-3

- Soland RM (1974) Optimal facility location with concave costs. *Operations Research* 22:373–382
- Stützle T, Hoos H (1997) Max-min ant system and local search for the traveling salesman problem. In: *IEEE International Conference On Evolutionary Computation (ICEC'97)*, IEEE Press, Piscataway,NJ, pp 309–314
- Sun M, Aronson JE, McKeown PG, Drinka D (1998) A tabu search heuristic procedure for the fixed charge transportation problem. *European Journal of Operational Research* 106:441–456
- Tuy H (2000) *Approximation and Complexity in Numerical Optimization: Continuous and Discrete Problems*, Kluwer Academic Publishers, chap The minimum concave-cost network flow problem with a fixed number of nonlinear arc costs: complexity and approximations, pp 383–402
- Tuy H, Ghannadan S, Migdalas A, Värbrand P (1995) The minimum concave cost flow problem with fixed numbers of nonlinear arc costs and sources. *Journal of Global Optimization* 6:135–151
- Zangwill W (1968) Minimum concave cost flows in certain networks. *Management Science* 14:429–450

Recent FEP Working Papers

474	Marta S.R. Monteiro, Dalila B.M.M. Fontes and Fernando A.C.C. Fontes, <i>Ant Colony Optimization: a literature survey</i> , December 2012
473	João Correia-da-Silva and Carlos Hervés-Beloso, <i>Irrelevance of private information in two-period economies with more goods than states of nature</i> , December 2012
472	Gonçalo Faria and João Correia-da-Silva, <i>Is Stochastic Volatility relevant for Dynamic Portfolio Choice under Ambiguity?</i> , October 2012
471	Helena Martins and Teresa Proença, <i>Minnesota Satisfaction Questionnaire - Psychometric Properties and Validation in a Population of Portuguese Hospital Workers</i> , October 2012
470	Pedro Mazedo Gil, Oscar Afonso and Paulo Brito, <i>Skill Structure and Technology Structure: Innovation and Growth Implications</i> , September 2012
469	Abel L. Costa Fernandes and Paulo R. Mota, <i>Triffin's Dilemma Again and the Efficient Level of U.S. Government Debt</i> , September 2012
468	Mariana Cunha and Vera Rocha, <i>On the Efficiency of Public Higher Education Institutions in Portugal: An Exploratory Study</i> , September 2012
467	Paulo R. Mota, Abel L. Costa Fernandes and Ana-Cristina Nicolescu, <i>The Recent Dynamics of Public Debt in the European Union: A Matter of Fundamentals or the Result of a Failed Monetary Experiment?</i> , September 2012
466	Elena Sochirca, Oscar Afonso and Sandra Silva, <i>Political rivalry effects on human capital accumulation and inequality: a New Political Economy approach</i> , September 2012
465	Mariana Cunha and Paula Sarmento, <i>Does Vertical Integration Promote Downstream Incomplete Collusion? An Evaluation of Static and Dynamic Stability</i> , August 2012

Editorial Board (wps@fep.up.pt)

Download available at: <http://wps.fep.up.pt/wplist.php>
also in <http://ideas.repec.org/PaperSeries.html>

www.fep.up.pt

UNDERGRADUATE, MASTER AND PHD PROGRAMMES



FACULDADE DE ECONOMIA
UNIVERSIDADE DO PORTO



School of Economics and Management, University of Porto
Rua Dr. Roberto Frias | 4200-464 Porto | Portugal
Telephone: +351 225 571 100