Locating and Sizing Bank-Branches by Opening, Closing or Maintaining Facilities

Marta S. Rodrigues Monteiro[1,2] and Dalila B. M. M. Fontes[2]

[1] DMCT - Universidade do Minho
Campus de Azurém, 4800 Guimarães, Portugal 020414011@fep.up.pt
[2] Faculdade de Economia - LIACC fontes@fep.up.pt

Summary. The bank-branch restructuring problem seeks to locate bank-branches by maintaining, closing, or opening branches, to provide the service required by clients, at minimum total cost. This nonlinear problem, due to the existence of economies of scale, is formulated as a mixed binary, integer linear model. The model obtained can be solved by a ready-available software. However, due to the problem combinatorial nature, only small size instances can be solved. Thus, we also propose a local search heuristic that iteratively improves the solution obtained for a related linear problem by applying drop and swap operations. The computational experiments performed show the effectiveness and efficiency of the proposed heuristic.

Keywords: Bank-branch, Location, Concave Optimization, Heuristics

1 Introduction

Although bank-branch restructures have long been present in the financial world, they have not been the subject of much academic study, particularly from the operational research point-of-view [6]. A similar problem is addressed by chance-constrained goal-programming in [1] where three levels of bank services are considered: ATM, branches, and main branches. In [2] the bank-branch location problem is addressed by a two stage procedure where the number of branches needed to provide the minimum coverage is found by solving a classical covering problem; and then, their exact location is determined by solving a maximal coverage location problem. A budget constrained facility relocation problem is studied in [3] where both opening and closing facilities is considered. Three heuristics were developed: greedy-interchange, tabu search, and lagrangean relaxation.

In this paper, a new heuristic based on local search is presented to solve the bank-branch restructuring problem. This heuristic is divided in two stages: (i) obtaining an initial solution by solving a related linear problem [5]; (ii) improving that solution by applying drop and swap operations. The rest of the paper is organized as follows: in section 2 we describe the bank-branch location and sizing problem considered in this work and give the mathematical model. In section 3 we explain the methodology used. Computational experiments are provided in section 4 and finally, in section 5 some conclusions are drawn.

2 Problem Definition and Mathematical Formulation

The bank-branch restructuring problem seeks to locate branches, such that client needs for banking services are satisfied at a minimum cost. This can be achieved by opening new branches, and closing or resizing existing ones. Client needs need not to be satisfied by a single branch. Costs are incurred by opening, closing, and operating branches, and by providing clients with the required service. For each client we consider an ideal coverage that must be satisfied, and a minimum coverage that may or may not be satisfied. A penalty cost is incurred whenever the coverage provided is below the ideal coverage. This cost is proportional to the difference between these values. Employees are also taken into account in our problem both in terms of costs (hiring and firing costs) and in terms of needs (branches require a pre-specified number of employees to be able to operate). As far as the authors are aware of this aspect has always been neglected in the literature. We consider that banks operate in different areas, named counties, and that each of these counties is divided into smaller regions, called parishes. We assume that all clients of a parish are located at its geographical centre. The same applies to branches thus, there can only exist a single branch per parish. Different branch sizes with different service capacity are considered.

Let C be the set of counties and D the set of parishes, where $D = \cup_j D_j$ with $j \in C$. Let also K be the set of branch sizes. Since we may take decisions on whether to open new branches and whether to close existing branches we have defined the following decision variables.

- $y_{ij}^k = \begin{cases} 1, \text{ if a branch of size } k \text{ is closed in parish } i \text{ of county } j, \\ \quad \text{where } j \in C, i \in CB_j, k \in K, \\ 0, \text{ otherwise.} \end{cases}$
- $z_{ij}^k = \begin{cases} 1, \text{ if a branch of size } k \text{ is opened in parish } i \text{ of county } j, \\ \quad \text{where } j \in C, i \in D_j \backslash NCB_j, k \in K, \\ 0, \text{ otherwise.} \end{cases}$
- $x_{ij}^k = \begin{cases} 1, \text{ if a branch of size } k \text{ is operating in parish } i \text{ of county } j, \\ \quad \text{where } j \in C, i \in D_j, k \in K, \\ 0, \text{ otherwise.} \end{cases}$
- $he_j \geq 0$, number of employees hired in county j
- $fe_j \geq 0$, number of employees fired in county j
- q_{ij}^{lm}, number of service units provided by branch in parish i of county j to client in parish l of county m.

$$\min \quad \sum_{j\in C}\sum_{i\in D_j}\sum_{k\in K} f_{ij}^{k}(x) + \sum_{j\in C}\sum_{i\in CB_j}\sum_{k\in K} g_{ij}^{k}(y) +$$
$$\sum_{j\in C}\sum_{i\in D_j\setminus NCB_j}\sum_{k\in K} h_{ij}^{k}(z) + \sum_{j\in C} T_j \times he_j + \sum_{j\in C} CMP_j \times fe_j +$$
$$\sum_{m\in C}\sum_{l\in D_m} P_{lm} \times (\overline{W}_{lm} - \sum_{j\in C}\sum_{i\in D_j} q_{ij}^{lm}) + \sum_{j\in C}\sum_{i\in D_j}\sum_{m\in C}\sum_{l\in D_m} q_{ij}^{lm} \times v_{ij}^{lm} \,. \tag{1}$$

subject to:

$$x_{ij}^{k_i} = 1, \quad \forall j \in C, \forall i \in NCB_j, k_i = k(i,j), \tag{2}$$
$$x_{ij}^{k\neq k_i} = 0, \quad \forall j \in C, \forall i \in NCB_j, \forall k \neq k_i \in K, \forall k_i = k(i,j), \tag{3}$$
$$x_{ij}^{k_i} = 1 - y_{ij}^{k_i}, \quad \forall j \in C, \forall i \in CB_j, k_i = k(i,j), \tag{4}$$
$$x_{ij}^{k\neq k_i} = z_{ij}^{k\neq k_i}, \quad \forall j \in C, \forall i \in CB_j, \forall k \neq k_i \in K, \forall k_i = k(i,j), \tag{5}$$
$$\sum_{k\in K} z_{ij}^{k} \leq 1, \quad \forall j \in C, \forall i \in D_j \setminus B_j, \tag{6}$$
$$x_{ij}^{k} = z_{ij}^{k}, \quad \forall j \in C, \forall i \in D_j \setminus B_j, \forall k \in K, \tag{7}$$
$$\underline{W}_{lm} \leq \sum_{j\in C}\sum_{i\in D_j} q_{ij}^{lm} \leq \overline{W}_{lm}, \quad \forall m \in C, \forall l \in D_m, \tag{8}$$
$$q_{ij}^{lm} \leq a_{ij}^{lm} \times \sum_{k\in K} k \times x_{ij}^{k}, \forall j, m \in C, \forall i \in D_j, \forall l \in D_m, \tag{9}$$
$$\sum_{m\in C}\sum_{l\in D_m} q_{ij}^{lm} \leq \alpha \times a_{ij}^{lm} \sum_{k\in K} k \times x_{ij}^{k}, \forall j \in C, \forall i \in D_j, \tag{10}$$
$$\sum_{j\in C}\sum_{i\in D_j}\sum_{k\in K} \epsilon_{ij}^{k}(x) \times x_{ij}^{k} = E + \sum_{j\in C} he_j - \sum_{j\in C} fe_j, \tag{11}$$
$$\sum_{i\in CB_j}\sum_{k\in K} \epsilon_{ij}^{k}(y) \times y_{ij}^{k} - fe_j \geq 0, \quad \forall j \in C, \tag{12}$$
$$\sum_{i\in D_j\setminus B_j}\sum_{k\in K} \epsilon_{ij}^{k}(z) \times z_{ij}^{k} - he_j \geq 0, \quad \forall j \in C, \tag{13}$$
$$he_j, fe_j, q_{ij}^{lm} \geq 0, \text{ integer, and } \quad x_{ij}^{k}, y_{ij}^{k}, z_{ij}^{k} \in \{0,1\} \,. \tag{14}$$

The objective function (1), minimizes the total cost, which is made up four components: branch costs (operating, closing, and opening costs); employee costs (hiring and firing costs); penalty costs; and service costs. The objective function is concave as it is given by the sum of linear and concave components (operating costs and service costs). The functions f_{ij}^{k}, g_{ij}^{k}, and h_{ij}^{k}, are non linearly dependent on several factors, see [6] for more details. Constraints (2) and (3) are related to the existing branches that are not allowed to be closed, while constraints (4) and (5) are related to the existing branches for which a closing decision is possible. Constraints (6) and (7) guarantee that at most one branch is opened at each new potential location and that it is operated. Constraint (8) guarantees that the service provided to each client is within the limits required, while constraints (9) and (10) are boundaries for the service provided by each branch to a single client and to all clients allocated to it, respectively. Constraints regarding the number of employees needed, fired, and hired are given by (11) to (13).

3 Solution Methodology

The above model has been set-up in a format such that CPLEX could be used to solve it. However, given that CPLEX works with matrices derived from the mathematical model that has $3 \times n_p \times n_c \times k$ binary variables and $2 \times n_c + n_c^2 \times n_p^2$ integer variables, the memory requirements are large and grow rapidly with problem size. Therefore, many of our problem instances could not be solved by CPLEX. In order to solve larger instances, which realistically banks are faced with, we have developed the following local search heuristic.

3.1 Initial Solution

In order to find an initial feasible solution we have solved a related linear programming problem that covers all demand locations at a minimum service cost. The objective function for this problem is,

$$\min \quad \sum_{j \in C} \sum_{i \in D_j} \sum_{m \in C} \sum_{l \in D_m} \phi_{ij}^{lm} \tag{15}$$

As before the service provided to each client must satisfy lower and upper limits, as in (8). The overall coverage capacity for each branch must be at most α times the maximum branch capacity if the branch is to be opened; or α times the existing branch capacity, otherwise. Similar constraints are imposed, but now to the covers that can be provided to a single client.

We successively solve this LP model with updated cost function ϕ_{ij}^{lm}. At each iteration the cost function is updated by using the information of the solution to the previous iteration. This approach is based on [5].

$$(\phi_{ij}^{lm})^T = \sum_{m \in C} \sum_{l \in D_m} ((v_{ij}^{lm})^T - P_{lm}) \times q_{ij}^{lm}.$$

Initially, we only consider the linear cost, i.e. $(v_{ij}^{lm})^0 = v_{ij}^{lm}$. The cost function is updated as follows:

$$(\phi_{ij}^{lm})^{T+1} = \begin{cases} (\overline{v}_{ij}^{lm})^T + \dfrac{h_{ij} + f_{ij}}{(q_{ij}^{lm})^T \times \varphi_{ij}^T}, & \text{if } (q_{ij}^{lm})^T > 0 \\ (\overline{v}_{ij}^{lm})^R, & \text{otherwise.} \end{cases}$$

where φ_{ij}^T is the number of demand locations serviced by branch in parish i of county j, at iteration T and R is the index of the last iteration where $(q_{ij}^{lm})^T > 0$. The update procedure stops whenever either the solution of two consecutive iterations is the same, or the maximum number of iterations is reached. The initial solution is provided by the best solution obtained at the end of the procedure.

3.2 Improving the Initial Solution

The initial feasible solution is improved further by consecutively applying the following steps.

Step 1 Attempt to drop branches that serve only one client, (a) as long as the minimum coverage is provided; (b) by distributing the service units provided to their clients by other branches still having available capacity.
Step 2 Try to eliminate branches which are not using all service capacity.
Step 3 Attempt to downsize branches, (a) as long as the minimum coverage is provided; (b) by distributing some service to other branches with available capacity.
Step 4 Try to swap branches of different locations.

We do not consider adding branches since typically the initial solution completely satisfies the ideal coverage. To compute the cost variation for each of the above steps all components of the original cost function must be included. Furthermore, the cost function to be used is the original.

4 Computational Experiments

The proposed local search heuristic has been implemented in Visual C++ 6.0. Computational experiments were carried out on a 1.8-GHz Pentium4 with 256 MB of RAM. The MIP model given in Sect. 2 has been implemented in CPLEX. The optimality gap is given by $Error = \frac{(x - \overline{x})}{\overline{x}} \times 100$. In Table 1 we report on the variation of the number of employees E; the percentage ratio Q between covers provided and ideal coverage; the number of operating branches B; and the computational time required to solve the problem, in CPU seconds, both for CPLEX and Heuristic. Overall 180 problems have been solved. For each entry of the table we report the number of parishes M, and the average number of counties N for the 30 problems we have generated. In average, the heuristic is quicker to solve a problem and, although the solution is usually more expensive it provides better service than the CPLEX solution, since more coverage is provided and more branches exist. As it can be seen in Fig. 1 the variation on the number of counties does not seem to affect the error, while the error gap increases with the number of parishes.

Table 1. Average quality of the solutions

m	n	E	Q%	B	Time	E	Q%	B	Time	Error%
			CPLEX				Heuristic			
15	13	-52	99	6	1	-45	98	7	1	6.24
25	16	-38	99	9	7	-30	99	10	3	5.99
35	23	-73	98	13	38	-63	100	14	3	3.81
45	37	-150	99	16	397	-138	100	18	5	3.42
55	36	-104	99	20	688	-92	100	22	8	4.16
65	16	55	97	25	1966	75	99	24	11	6.18
Average		**-60**	**99**	**15**	**516**	**-49**	**99**	**16**	**5**	**5**

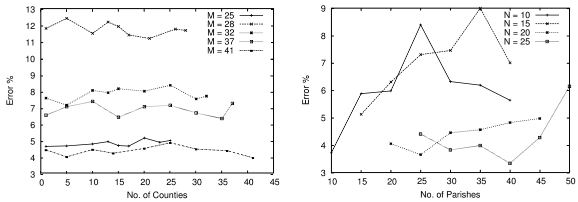

Fig. 1. Average error for varying number of (**a**) Counties (**b**) Parishes

5 Conclusion

We have developed a local search heuristic to solve the bank-branch location and sizing problem with concave cost functions. The heuristic is based on the solution to a related linear integer programming problem, iteratively improved by applying drop and swap operations. The computational experiments indicate that our heuristic is faster and that the number of counties does not affect the solution. The number of parishes affects the optimality gap due to the combinatorial nature of the problem.

References

1. Min H, Melachrinoudis E (2001) The three-hierarchical location-allocation of banking facilities with risk and uncertainty. Int Trans Oper Res 8:381-401
2. Miliotis P, Dimopoulou M, Giannikos I (2002) A hierarchical location model for locating bank branches in a competitive environment. Int Trans Oper Res 9:549-565
3. Wang Q, Batta R, Bhadury J, Rump CM (2003) Budget constrained location problem with opening and closing of facilities. Comput Oper Res 30:2047-2069
4. Kim D, Pardalos PM (1999) A Solution Approach to the Fixed Charge Network Flow Problem Using a Dynamic Slope Scaling Procedure. Oper Res Lett 24:195-203
5. Kim D, Pardalos PM (2000) Dynamic slope scaling and trust interval techniques for solving concave piecewise linear network flow problems. Networks 35(3):216-222
6. Monteiro MSR (2005) Bank-branch location and sizing under economies of scale. Master Thesis, Faculdade de Economia do Porto, Portugal