Computational complexity and decomposition algorithms for the mill pricing problem

A. V. Plyasunov*, A. A. Panin†

*Sobolev Institute of Mathematics, apljas@math.nsc.ru
†Novosibirsk State University, arteam1891@gmail.ru

In the mill pricing problem (MP) we are given two finite sets: a set of facilities and a set of customers. Each customer has a budget and a demand. He patronizes a facility providing the lowest sum the travel cost to the facility and price for its product. The objective is to determine the price for each facility to maximize the overall revenue. We present the problem as a linear integer programming problem and study its computational complexity. Exact and approximate algorithms are developed for this problem. Computational results for random generated test instances are discussed.

Mathematical model

Let us introduce the following notations:
$I = \{1, ..., n\}$ is the set of facilities;
$J = \{1, ..., m\}$ is the set of customers;
$b_j \geq 0$ is the budget of customer j;
$c_{ij} \geq 0$ is the travel cost for each pair of customer j and facility i.

Now we define the decision variables:

$p_i \geq 0$ is the price of facility i;

$$x_{ij} = \begin{cases} 1 \text{ if customer } j \text{ patronizes facility } i, \\ 0 \text{ otherwise.} \end{cases}$$

Using these variables we may present the problem as a mixed integer quadratic program:

$$\max_{p,x} \sum_{i \in I} p_i \sum_{j \in J} x_{ij} \tag{1}$$

$$\sum_{i \in I} (b_j - c_{ij} - p_i) x_{ij} \geq 0, \quad i \in I, j \in J; \tag{2}$$

$$\sum_{i \in I} (c_{ij} + p_i) x_{ij} \leq c_{kj} + p_k, \quad i, k \in I, j \in J; \tag{3}$$

$$\sum_{i \in I} x_{ij} \leq 1, \quad j \in J; \tag{4}$$

$$p_i \geq 0, x_{ij} \in \{0, 1\}, \quad i \in I, j \in J. \tag{5}$$

The objective function (1) defines the total revenue of the facilities. Constraints (2) ensure that each customer does not exceed his own budget. Constraints (3) guarantee that each customer patronizes a facility providing the lowest sum the travel cost to the facility and price for its product. Constraints (4) ensure that each customer can be served by at most one facility. The following theorems characterize the computational complexity of the problem.

Theorem 1. *[1, 2] The MP problem is strongly NP-hard.*

Theorem 2. *The problem MP belongs to the class log-APX.*

Theorem 3. *If $P \neq NP$, then no polynomial time absolute approximation algorithm exists for the MP problem.*

Let us consider the following decision problem $Opt(MP)$. For given value k and given input data of the MP problem, we need to check whether the optimal value of the problem is k. This decision problem belongs to the class Δ_2^P [3]. The following theorem claims that $Opt(MP)$ problem is more difficult than arbitrary problem from the classes NP and co-NP.

Theorem 4. *If $NP \neq co-NP$, then the problem $Opt(MP)$ is a proper-Δ_2^P problem.*

Linear programming reformulation

Let $\overline{p}_i$ be the most possible price of facility i:

$$\overline{p}_i := \max_j (b_j - c_{ij}).$$

We introduce new decision variables z_{ij} as the revenue of facility i from customer j:

$$z_{ij} := p_i x_{ij}.$$

Now we can rewrite the MP problem as a linear mixed integer program:

$$\max_{p,x,z} \sum_{i\in I}\sum_{j\in J} z_{ij} \quad (6)$$

$$\sum_{i\in I}(b_j - c_{ij})x_{ij} - \sum_{i\in I} z_{ij} \geq 0, \quad j \in J; \quad (7)$$

$$c_{kj} + p_k \geq \sum_{i\in I} c_{ij}x_{ij} + \sum_{i\in I} z_{ij},\ k \in I,\ j \in J; \quad (8)$$

$$(1 - x_{ij})\overline{p}_i - z_{ij} + p_i \geq 0, \quad i \in I,\ j \in J; \quad (9)$$

$$(1 - x_{ij})\overline{p}_i + z_{ij} - p_i \geq 0, \quad i \in I,\ j \in J; \quad (10)$$

$$z_{ij} \leq \overline{p}_i x_{ij}, \quad i \in I,\ j \in J; \quad (11)$$

$$\sum_{i\in I} x_{ij} \leq 1, \quad j \in J; \quad (12)$$

$$p_i \geq 0, x_{ij} \in \{0,1\}, z_{ij} \geq 0, \quad i \in I,\ j \in J. \quad (13)$$

For given x, the problem (6)–(13) is a linear programming problem. Let λ be the vector of dual variables for constraints (7) – (12). Denote by $\delta(x,\lambda)$ the objective function of the dual problem $D(x)$ which can be obtained from the problem (6) – (13) for given x. Moreover, denote by $\delta_{ij}^1(x,\lambda)$ and $\delta_i^2(x,\lambda)$ the constraints of the dual problem which are corresponding to variables z_{ij} and p_i. Below we apply the Benders decomposition approach and present an exact hybrid decomposition method [4, 5].

The Basic method

<u>**Step 1**</u>: Apply an approximate algorithm for the problem (6) – (13) in order to get a family of feasible solutions $x^r, r = \overline{1,R}$. For each solution we solve the following problem:

$$\rho(x^r) = \min_\lambda \delta(x^r,\lambda)$$
$$\delta_{ij}^1(x^r,\lambda) \leq 0,\ i \in I, j \in J;$$
$$\delta_i^2(x^r,\lambda) \leq 0,\ i \in I.$$

If the problem is solvable, then we find the optimal values of dual variables λ^r. Otherwise, we define λ^r as the direction vector of an infinite edge. Let $LB := \max\{\delta(x^r,\lambda^r), r = \overline{1,R}\}$. It is a lower bound for (6).

<u>**Step 2**</u>: Solve the relaxed master problem:

$$\max_{x_{ij}\in\{0,1\}, y\geq 0} y$$
$$y \leq \delta(x,\lambda^q), q = \overline{1,Q};$$
$$\delta(x,\lambda^u) \geq 0, u = \overline{1,U};$$
$$1 - \sum_{i\in I} x_{ij} \geq 0,\ j \in J,$$

where U is the number of the direction vectors for the infinite edges, and Q is the number of corresponding vertices. Let $(\overline{y},\overline{x})$ be the optimal solution, then $UB := \overline{y}$ is an upper bound.

<u>**Step 3**</u>: Solve the subproblem $\rho(x)$ with $x = \overline{x}$.
Case 1. The subproblem is solvable. If $UB = \rho(\overline{x})$, then stop, the optimum is found. Otherwise we put $Q := Q + 1, \lambda^Q := \lambda$, where λ is the optimal values of dual variables. If $\rho(\overline{x}) > LB$, then $LB := \rho(\overline{x})$ is the new lower bound. Go to Step 2.
Case 2. The subproblem is unsolvable. We define λ as the direction vector of an infinite edge and put $U := U + 1, \lambda^U := \lambda$. Go to Step 2.

This algorithm is finite but shows slow convergence [4–9]. Therefore, in this paper we suggest new decomposition schemes to accelerate the convergence to global optimum. In [4, 10] the randomized metaheuristics were applied for this end at Step 1. In this paper we propose some two-phase decomposition methods. The main idea is following. At the first stage, we solve the linear programming relaxation of (6) – (13) by a hybrid decomposition algorithm and use continuous variables x_{ij} at Step 2 in the relaxed master problem. At the second stage, we consider the mixed integer problem (6) – (13) and include the optimal family of cuts from the first stage into the family of feasible solutions $x^r, r = \overline{1, R}$ at Step 1 of the Basic method.

Actually, we consider the following algorithmic schemes:

1. Basic scheme.
2. Two-phase scheme.
3. Scheme 2 with an approximate algorithm at the first phase.
4. A simplified scheme 2.
5. Scheme 2 with control of constraints.

The schemes 1,2,3,5 are exact, the scheme 4 is approximate.

Computational results

These five schemes were coded in Delphi 7.0 environment and tested on random generated test instances. For all experiments the values of parameters b_j and c_{ij} are taken from the interval [1, 99] at random with uniform distribution. Tables 1 – 4 show the results of experiments for $n \leq 50$, $m \leq 50$. As we can see, the scheme 2 is the fastest exact scheme, and scheme 1 is the slowest.

Table 1. Running time of schemes 1, 2

		Scheme 1	Scheme 2
n	m	*time*	*time*
5	10	23"	10.2"
5	15	302"	30.2"

Table 2. Running time of scheme 3

		Deviation at the first stage			
n	m	ε=0.02	ε=0.05	ε=0.10	ε=0.20
5	10	9.4"	9.2"	8.6"	12"
5	15	40.2"	57.8"	71.6"	164.8"

Table 3. Running time of scheme 5

		Number of constraints		
n	m	20	30	60
5	10	15.6"	17.2"	12.6"
5	15	85.8"	82.4"	71.2"

In the scheme 3 we modify the first phase of the method. We break the computation at the first stage if the inequality $(UB - LB)/UB < \varepsilon$ holds. Computational experiments were carried out for the following values of parameter $\varepsilon = 0.02, 0.05, 0.1, 0.2$. The scheme 3 was always faster than scheme 1. It is interesting that the running time grows when ε increases. Thus the correct choose of the parameter can accelerate the computations. Note that scheme 3 is an improved version of the approach from [8].

The idea of the scheme 5 is to control the number of cuts. For this end we use three lists: two lists with fixed lengths (Q and U) and the third list with an arbitrary length. The first list with length Q is used to store the optimal cuts. The second list with length U is used to storage the feasibility cuts. When a new cut is added to corresponding list, the oldest cut is dropped and placed in the unrestricted list. If new upper bound, defined the current relaxed master problem, is less than the previous value then we return back the dropped cut. Computational experiments were carried out with a total number of optimality and feasibility cuts equals to 20, 30 and 60. As we can see from Table 3, the running time decreases when the total number of cuts grows. This scheme is better than scheme 1, but worse than scheme 2. Therefore, it is important to achieve high quality of approximation in the first phase to reduce the total running time of the two-phase scheme.

Table 4. Running time of scheme 4

n	m	ε_1	ε_2	*time*
5	10	0.0651	0.0651	5.4"
5	15	0.0686	0.0686	9"
5	30	0.123	0.123	146.8"
5	50	0.196	0.196	1639.6"
10	20	0.114	0.114	116.2"
10	30	0.151	0.145	1194.6"
20	20	0.106	0.106	1164.6"
20	40	0.334	0.298	3885"
30	30	0.249	0.217	3105.6"

In experiments with the approximate scheme 4, the quality of approximation in the first phase helps us to improve upper bound. In scheme 4 calculations terminate at Step 2, the second iteration of the second phase. Table 4 presents computational results of our experiments for the scheme. The third and fourth columns present the deviations $\varepsilon_1 = (UB - LB)/UB$ and $\varepsilon_2 = (UB - Opt)/UB$.

The fifth column shows the running time. Table 4 shows that the time complexity of this scheme is significantly less than the time complexity of other schemes. Moreover, the deviation from the optimum is not large.

Список литературы

[1] V. T. Dementyev, Yu. V. Shamardin *Problem of a choice of the prices for production with obligatory satisfaction of demand.* Discrete Analysis and Operations Research. 2002. Ser. 2. Vol 9. № 2. P. 31-40.

[2] Panin A.A. *Genetic algorithm for one problem of pricing.* Proceedings ICM&MG SB RAS Ser. Informatics. 2009. Issue 9. P. 190-196.

[3] M. R. Garey, D. S. Johnson, *Computer and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, CA, 1979.

[4] A. V. Plyasunov, *Hybrid Methods Using Decomposition for the Hard Combinatorial Optimization Problems.* Proceedings of the 8th International Conference on Intellectualization of Information Processing, Cyprus, Paphos, 17-24 October, 2010. - P. 286–289.

[5] A. V. Geoffrion *Generalized Benders decomposition* Journal of Optimization Theory and Application. 1972. Vol. 10, No. 4. P. 237–260.

[6] F. Vanderbeck, L. A. Wolsey *Reformulation and decomposition of integer programs.* Monreal, 2009. CORE DP Vol. 16.

[7] T. L. Magnanti, R. D. Wong *Decomposition methods for facility location problems* Mirchandani P.D., Francis R.L. (Eds). Discrete Location Theory. - Wiley and Sons, 1990. - P. 439-478.

[8] D. McDaniel, M. Devine *A modified Benders partitioning algoritm for mixed integer programming* Management Science. 1977. Vol. 24. P. 312-319.

[9] G. Gote, M. A. Laughton *Large scale mixed integer programming: Benders-type heuristics* European Journal of Operational Research. 1984. Vol. 16. P. 327-333.

[10] E. V. Alekseeva, N. A. Kochetova, Yu. A. Kochetov, A. V. Plyasunov *A heuristic and exact methods for the dicrete $(r|p)$-centroid problem.* LNCS, Berlin: Springer, 2010. V. 6022. P. 11-22.