A Second Order-Cone Programming Relaxation for Facility Location Problem and Simple Assembly Line Balancing Problem

Moraru V., Zaporojan S.
Technical University of Moldova
Chisinau, Republic of Moldova
moraru@mail.utm.md, zaporojan@mail.utm.md

Cojocari N.
State University „Alecu Russo"
Balti, Republic of Moldova
cojocari.natalia@gmail.com

Abstract **– The paper put into discussion the numerical solving of the well known problems in the field of production planning - the facility location problem and the balancing problem SALBP-I (Simple Assembly Line Balancing Problem - I). A model in terms of Second-Order Cone Programming (SOCP) of the above problems is obtained.**

Key words **– Facility location problem, SALBP, Second Order-Cone Programming, Relaxation.**

I. Introduction

The well known problems in the area of production planning are defined as:

- the facility location problem,
- assembly line balancing problem.

Both of them play an important role in the modern economical world. The investigation in this field is presently attracting much attention because of their impact on optimal organization of the production. The facility location problem involves grouping of equipment, machinery, spaces available, etc. in order to determine the best solutions to an objective function, while the industrial information system requirements are satisfied [1,2]. On the other hand, the assembly industry is a promising way of balancing assembly lines to quickly adapt to possible changes [3].

Most of mathematical models for the considered problems are formulated in terms of linear programming with binary variables and fit into the class of NP-hard problems [4, 5]. It is probably impossible to secure optimal solutions using fast algorithms, in the case of NP-hard problems. It is really difficult to apply the methods for large scale problems. Therefore, approximate methods are used by which solutions can be found "good" in a reasonable time. It is well known, that for NP-hard problems an approximation is constructed for the whole problem. When building a specific approximation, different methods and techniques may be used together, such as exact polynomial methods, iterative approaches, and relaxation methods. Approximate methods fall into two broad classes: heuristics and local search methods based on relaxation of the formulated problems, renouncing provided that problems are integer variables [6]. In this paper we propose reformulation and relaxation in terms of second-order cone programming (SOCP) of the considered problems. Such type of relaxation supposes that the variable $y_i = 0$ or $y_i = 1$ is equivalent to condition $y_i = y_i^2$. Similar relaxation techniques were proposed in [7, 8].

The paper is organized as follows. In section II, we briefly describe mathematical models for considered problems. In section III, we show that it is possible an SOCP relaxation for facility location problem and simple assembly line balancing problem. In section IV, we conclude with a few final remarks.

II. Mathematical Programming Models

A. The Facility Location Problem

Let

$D = \{1, 2, \ldots, s\}$ - a finite set of "clients";

$F = \{1, 2, \ldots m\}$ - a finite set of possible "facilities";

$f_i \in \Re_+$ for every facility $i \in F$;

$c_{ij} \in \Re_+$ - the costs of service for ;

$x_{ij} = 1$, if facility *i* serves the client *j*, otherwise $x_{ij} = 0$;

$y_i = 1$, if the facility *i* is open, otherwise $y_i = 0$.

The mathematical model of the facility location problem can be expressed as [1, 2, 6]:

$$\left.\begin{array}{l} \sum\limits_{i \in F} f_i y_i + \sum\limits_{i \in F} \sum\limits_{j \in D} c_{ij} x_{ij} \to \min \\ x_{ij} \le y_i, i \in F, j \in D, \\ \sum\limits_{i \in F} x_{ij} = 1, j \in D, \\ x_{ij}, y_i \in \{0,1\}, i \in F, j \in D. \end{array}\right\} \qquad (1)$$

The objective function minimizes the sum of cost and fixed costs for establishing facilities. The constraints $x_{ij} \le y_i$ ensure that every location $j \in D$ is assigned to some location $i \in F$.

B. The Simple Assembly Line Balancing Problem

The assembly line balancing problem was first formulated in 1955 by Salvenson [3]. The general problem of assembly line balancing can be defined in various ways. The problem definition is as follows. The assembly line consists of a finite number of workstations that are running individual operations (tasks) to manufacture a product. The problem now is how to combine operations and workstations in order to obtain an optimal distribution of the workload to a minimum number of stations. At the same time, it is necessary to ensure conditions of precedence for the execution of operations. There were defined a few types of the assembly line balancing problem.

The first class of problem (type I or SALB-I) minimizes the number of workstations, maintaining the desired cycle time [10].

We denote by P the immediate precedence matrix of dimension $n \times n$:

$$P(i, j) = \begin{cases} 1, \text{if task } j \text{ is an immediate successor of task } i, \\ 0, \text{otherwise.} \end{cases}$$

It is considered to be known the production rate R (the number of items collected per unit time) and the processing time t_i of the operation i. The number of stations can not be lower than T/C, where $T = \sum_{i=1}^{n} t_i$ is the time of all needed operations, and $C = 1/R$ is the total cycle time of the assembly line. It is required to perform the distribution of operations to workstations so that the number of jobs to be minimal, i.e. the maximizing of the number of "empty" jobs. An "empty" is a plant that does not perform any operation.

It is assumed an assembly line of *n* stations. At each station runs one operation by a person or an automatic device (robot), which is needed to manufacture a product. For product assembly all operations must be done in a strict order. Assume the following conditions [11]: the assembly line is designed for a single product and supports only one mode of functioning; the stations are serial arranged; the execution time for operations is deterministic; the partition of operations is prohibited; all operations must be performed; there are precedence constraints; the execution time of an operation does not depend on the station on which is running; the cycle time is fixed.

We introduce Boolean variables x_{ij} and y_i:

$$x_{ij} = \begin{cases} 1, \text{if task } i \text{ is assigned to workstation } j, \\ 0, \text{otherwise.} \end{cases}$$

$$y_i = \begin{cases} 1, \text{if workstation } i \text{ has a task assigned to it,} \\ 0, \text{otherwise.} \end{cases}$$

such that, $x_{ij} = 1$ if the operation i is carried out at the workstation j and $x_{ij} = 0$ when it is carried out at another station; $y_i = 1$ if one of the operations is performed at the station i, and $y_i = 0$ in the opposite case. For the *SALBP-I* problem different formulations have been proposed. We will consider the following mathematical model [11]:

$$\sum_{i=1}^{n} y_i \to \min \tag{2}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = 1, \forall i = 1,2,\ldots,n, \tag{3}$$

$$\sum_{i=1}^{n} t_i x_{ij} \le C, \forall j = 1,2,\ldots,n, \tag{4}$$

$$x_{ik} \le \sum_{j=1}^{k} x_{sj}, \forall k = 1,2,\ldots,n, \forall i,s : P(i,s) = 1, \tag{5}$$

$$\sum_{i=1}^{n} x_{ik} \le n(1 - y_k), \forall k = 1,2,\ldots,n, \tag{6}$$

$$x_{ij} \in \{0,1\}, y_i \in \{0,1\}, \forall i, j = 1,2,\ldots,n. \tag{7}$$

Constraints (3) ensure that each operation will be performed only at a single workstation, and constraints (4) - that the cycle must be greater than or equal to the length of time at all stations. Constraints (5) require the precedence relations between operations. If $x_{ik} = 0$ (operation *i* is not running at the station *k*), then $\sum_{j=1}^{k} x_{ik}$ may take any value of 0 or 1 and constraints (5) become $\sum_{j=1}^{n} x_{sj} \ge 0$, it is always true and not represent a constraint. If $x_{ik} = 1$, then constraints (5) are equivalent to restrictions (3). Constraints (6) mean the following: if $y_k = 0$, then $\sum_{i=1}^{n} x_{ik} \le n$, which relationship is always satisfied. Thus (2) gives the number of "empty" jobs, i.e. the number of stations with $\sum_{i=1}^{n} x_{ik} = 0$, stations that do not perform any operation. Constraints (7) force x_{ij} and y_i to be binary.

Various heuristic and exact methods [11-14] have been proposed to solve the zero-one linear program (2)-(7).

III. A SECOND ORDER-CONE PROGRAMMING RELAXATION

We use the following notation in this paper:

$x^T y = \sum_i x_i y_i$ for the inner product of column vectors *x* and *y*;

x^T denotes the transposition of x and x^T is row vector;

$\|x\|_2 = \sqrt{x^T x}$ – the Euclidean norm of a vector *x;*

x_i denotes the i^{th} component of *x*;

e_i – the vector with all components equal to zero, except the i^{th} component, which is equal to one;

I – the identity matrix.

The condition that y_i and x_{ij} are binary is equivalent to the non-convex quadratic constraints:

$$y_i^2 - y_i = 0 \text{ and } x_{ij}^2 - x_{ij} = 0,$$

which in turn are equivalent to the following constraints:

$$\left.\begin{array}{c} y_i^2 - y_i \le 0, \\ \sum_i (y_i - y_i^2) \le 0, \end{array}\right\} \text{ and } \left\{\begin{array}{c} x_{ij}^2 - x_{ij} \le 0, \\ \sum_i \sum_j (x_{ij} - x_{ij}^2) \le 0. \end{array}\right.$$

Define $M_i = e_i e_i^T$ – the matrix whose all entries are zero, except the (i,i) entry which is one.

The constraint $y_i^2 - y_i \le 0$ can thus be reformulated as:

$$y^T M_i y - e_i^T y \le 0,$$

from which

$$y^T (M_i + I) y - e_i^T y - y^T y \le 0, \tag{8}$$

where

$$y = (y_1, y_2, \ldots, y_n)^T.$$

It can be easily observed that the matrix $M_i + I$ is positive definite, that you can rewrite using Cholesky decomposition as:

$$M_i + I = L_i^T L_i = L_i^2$$

where

$$L_i = \sum_{s \ne i} e_s e_s^T + \sqrt{2} e_i e_i^T .$$

Let nonnegative variable

$$t_0 = y^T y \ge 0.$$

Then the relationship (8) can be rewritten as:

$$\omega_i^T \omega_i \le \xi_i \eta_i \tag{9}$$

where

$$\omega_i = L_i^T y = \sum_{s \ne i} y_s e_s + \sqrt{2} y_i e_i ,$$

$$\xi_i = 1 \text{ and } \eta_i = e_i^T y + t_0 \ge 0$$

The hyperbolic constraint (9) is equivalent to the second-order cone constraints [9]:

$$\left\| \begin{pmatrix} 1 - e_i^T y - t_0 \\ 2 L_i y \end{pmatrix} \right\|_2 \le 1 + e_i^T y + t_0 .$$

Analogously can be obtained the second-order cone reformulation for the constraints

$$x_{ij}^2 - x_{ij} \le 0 .$$

For this we denote the vector:

$$X_j = (x_{1j}, x_{2j}, \ldots, x_{nj})^T .$$

It can be established that the constraints $x_{ij}^2 - x_{ij} \le 0$ are equivalent with:

$$X_j^T M_i X_j - e_i^T X_j \le 0, \forall i, j ,$$

or

$$X_j^T (M_i + I) X_j - e_i^T X_j - X_j^T X_j \le 0, \forall i, j .$$

Introducing the variables

$$t_j = X_j^T X_j \ge 0,$$

and applying the above technique, we obtain

$$\left\| \begin{pmatrix} 1 - e_i^T X_j - t_j \\ 2 L_i X_j \end{pmatrix} \right\|_2 \le 1 + e_i^T X_j + t_j, \forall i, j . \tag{10}$$

On the other hand, it can be observed that the conditions

$$\sum_i (y_i - y_i^2) \le 0 \text{ and } \sum_i \sum_j (x_{ij} - x_{ij}^2) \le 0$$

implies the linear constraints:

$$\left.\begin{array}{l} \sum_i y_i - t_0 \le 0, \\ \sum_j X_j - t_j \le 0, \forall j. \end{array}\right\} \quad (11)$$

Finally, we mention that the quadratic constraints

$$t_0 = y^T y, \;\; t_1 = X_1^T X_1, t_2 = X_2^T X_2, \ldots, t_n = X_n^T X_n \quad (12)$$

used above in constructing second-order cones (9) and (10) are non-convex. We relax constraints (12) as follows:

$$y^T y \le t_0, X_j^T X_j \le t_j, \forall j$$

The last constraints are equivalent to following second order cones constraints:

$$\left.\begin{array}{l} \left\| \begin{pmatrix} 1 - t_0 \\ 2y \end{pmatrix} \right\|_2 \le 1 + t_0, \\ \left\| \begin{pmatrix} 1 - t_j \\ 2X_j \end{pmatrix} \right\|_2 \le 1 + t_j, \forall j \end{array}\right\} \quad (13)$$

Thus, we may relax the problem (2)-(6) to the following second-order cone programming problem:

$$\sum_{i=1}^{n} y_i \to \min$$

subject to (3)-(7), (10), (13) and

$$e_i^T y + t_0 \ge 0, i = 1,2, \ldots n,$$
$$e_i^T X_j + t_j \ge 0, i = 1,2, \ldots n, j = 1,2, \ldots n.$$

IV. Conclusions

The paper presents a relaxation of the simple assembly line balancing problem and facility location problem in terms of second-order cone programming. These problems can be effectively solved by the interior point algorithm [15]. There is specialized software for solving conic optimization problems [16].

References

[1] C. S. ReVelle, H. A. Eiselt. Location analysis: A synthesis and survey, European Journal of Operational Research, 2005, Vol. 165 (1), pp. 1-19.

[2] В. Л. Береснев, Э. Х. Гимади, В.Т. Дементьев. Экстремальные задачи стандартизации. Новосибирск: Наука, 1978.

[3] M. E. Salvenson. The Assembly Line Balancing Problem. Journal of Industrial Engineering, 1955, Vol. 6, No. 4, pp.18-25.

[4] B. Korte, J. Vygen. Combinatorial Optimization: Theory and Algorithms. Springer, Berlin 2000 (Second edition 2002).

[5] T. K. Bhattachaje, S. Sahn. Complexity of Single Model Assembly Line Balancing Problems. Engineering Cost and Production Economics, 1990, 18, pp. 203-214.

[6] P. Mirchandani P., Francis R., eds. Discrete location theory. John Wiley and Sons, Inc., New York, 1990.

[7] V. Moraru, S. Zaporojan. A second order-cone programming formulation for simple assembly line balancing problem. Proceeding of the Third International Conference "Modelling and Development of Intelligent Systems", October 10-12, 2013, Sibiu, ROMANIA, ISSN 2067-3965, pp. 75-79, Lucian Blaga University Press, Sibiu, 2014.

[8] V. Moraru, S. Zaporojan, A. Groza. A second order-cone programming relaxation for Facility Location Problem. Proceeding of the 12th International Conference "Development and Application Systems", May 15-17, 2014, Suceava, ROMANIA, pp.189-191. Publisher: IEEE.

[9] F. Alizadeh, D. Goldfarb. Second-Order Cone Programming. Math. Program. 2003, Vol. 95, No1, Ser. B, pp.3-51.

[10] A. School. Balancing and Sequencing of Assembly Lines (series: Contributions to Management Science). Physica-Verlag Heidelberg, 2nd edition, 1999, 318 p.

[11] I. Baybars. A survey of exact algorithms for the simple line balancing problem. Management Science, 1986, 32, pp. 909-932. http://www.jstor.org/stable/2631657.

[12] A. School, C. Becker. State of the Art Exact and Heuristic Solution Procedures for Simple Assembly Line Balancing. European Journal of Operational Research, Vol. 168, No. 3, pp. 666-693, 2006.

[13] A. L. Arcus. COMSOAL a Computer Method of Sequencing Operations for Assembly Lines. The International Journal of Production Research, Vol. 4, No. 4, pp. 259-277, 1996.

[14] E. Erel, S. C. Sarin. A Survey of the Assembly Line Balancing Procedures. Production Planning and Control, Vol. 9, No. 5, pp. 414-434, 1998.

[15] Yu. Nesterov, A. Nemirovsky. Interior-point polynomial methods in convex programming. Volume 13 of Studies in Applied Mathematics. SIAM, Philadelphia, PA, 1994.

[16] H. D. Mittelmann. The State-of-the-Art in Conic Optimization Software. In Handbook on Semidefinite, Conic and Polynomial Optimization, Volume 166, pp. 671-686, 2012.