Transfer Line Balancing Problem

Transfer Line Balancing Problem (TLBP) was introduced in [3] In comparison with the well-known assembly line balancing problems, TLBP has many new assumptions reflecting the particularities of the machining environment. Machining transfer lines are usually paced and serial. They consist of a sequence of stations linked by an automated material handling device. In lines of this type, each station is equipped by a special machine-tool which performs machining operations block by block. All operations of each block are executed simultaneously using one multi-spindle head. The parallel execution of operations is possible due to the fact that the multi-spindle heads carry several simultaneously activated tools. When machining at the current station is finished (all blocks installed on this machine have been activated) the part is moved to the next station. The time span between two movements can not exceed the given time value T0 referred to as line cycle time. The balancing problem consists in assigning the given set of operations to parallel blocks and stations under given assignment restrictions.

The objective is to configure a serial machining line that consists of a number of linearly ordered machines. All the machines are linked by an automated material handling device without buffers between machines. Each machine is equipped with multi-spindle heads activated sequentially by changing the active spindle head or by moving the part to successive spindle heads. These multi-spindle heads perform different machining operations on the manufactured part. This type of equipment allows performing simultaneously several operations by tools fixed in the spindle head and activated simultaneously. The part transport is synchronized: all parts of the line are transferred simultaneously to the next station.

The line balancing problem in the assembly environment is well-studied in the literature (see e.g. [1]). The TLBP has a number of unique characteristics such as parameterized operation times, non-strict precedence constraints, and parallel operations execution. These features make it impossible to use directly the optimization methods developed for the Assembly Line Balancing Problems. Several exact (e.g. mixed-integer programming and graph approaches) and heuristic (e.g. FSIC and multi-start decomposition algorithm) methods have been developed for the TLBP, a summarized description of these methods is given in [6]. Later, in [7, 2] it was proposed to use greedy randomized adaptive search procedure (GRASP) and a genetic algorithm (GA) for solving this problem.

Notation

Let $\boldsymbol{N}$ ($i \in \boldsymbol{N}$) be the set of all operations i required to machine a part. Then, a solution of TLBP can be represented by a collection $P = \{\{N_{11},...,N_{1n_1}\},...,\{N_{m1},...,N_{mn_m}\}\}$, determining an assignment of operations to a sequence of machines (k = 1, 2, …, m) and repartition of operations assigned to the same machine k to n_k blocks. Let N_k be the work content (i.e. a set of operations) for machine k (k = 1, 2, …, m) and N_{kl} be the set of operations grouped into common block l (l =1, 2, …, n_k) of machine k.

To distinguish the pieces of equipment and corresponding sets of operations, the following notions are used: “station” for a set of operations assigned to a machine and “block” for a set of operations assigned to a spindle head.

The following input data is assumed to be known about the set of operations:

- **Precedence constraints** between the operations. The technological process defines a partial order relation over operation set $\boldsymbol{N}$. They can be represented by digraph $G = (\boldsymbol{N}, D)$. The arc $(i, j) \in \boldsymbol{N}\times\boldsymbol{N}$ belongs to set D if and only if operation j cannot precede operation i. Such precedence constraints are called non strict because if operation i is a predecessor of operation j then either operation i can be executed before j or i and j can be executed simultaneously using a compound tool;

- **Inclusion constraints** containing the groups of operations that must be assigned to the same machine, because of a required machining tolerance. They are can be represented by ES, a family of subsets from $\boldsymbol{N}$**,** such that all operations of the same subset $e \in ES$ must be assigned to the same station. No two sets from ES can include the same operation; otherwise such sets should be united.

- **Station exclusion constraints** containing the groups of operations that cannot be assigned to the same machine because of their technological incompatibility. They can be represented by $\overline{ES}$, a family of subsets from $\boldsymbol{N}$**,** such that each subset $e \in \overline{ES}$ cannot be assigned to the same station as a whole.

- **Block exclusion constraints** containing the groups of operations that cannot be assigned to the same spindle head because of their technological incompatibility. They can be represented by $\overline{EB}$, a family of subsets from $\boldsymbol{N}$**,** such that the same subset $e \in \overline{EB}$ cannot belong to the same block as a whole.

The following input data describes the machinery that can be used in the line design:

- τ^b is an auxiliary time needed for activation of a spindle head;

- τ^S is an auxiliary time needed for loading/unloading the part on a machine;

- C_1 is the relative cost of one station;

- C_2 is the relative cost of one block;

- n_0 is the maximal number of blocks per station;

The following data describes the line constraints to be respected:

- m_0 is the maximal authorized number of stations; this parameter is introduced in order to reduce the number of feasible solutions. Its value can be obtained by analyzing the available factory area or the maximum authorized line investment cost, or calculated as an upper bound on m.

- T_0 is the maximal admissible line cycle time (desired productivity).

To calculate the line cycle time that represents the time delay between the production of two products and to verify the line cycle constraint, the following calculation has to be made:

- For each operation j, its required working stroke length λ_j and the maximal admissible feed per minute s_j are given. The working stroke length includes the required depth of cut and the distance between the tool and the part surface.

- The block processing time $t^{b}(N_{kl})$ is determined as follows:

$$t^{b}(N_{kl}) = \max\{\lambda_j \mid j \in N_{kl}\} / \min\{s_j \mid j \in N_{kl}\} + \tau^{b}.$$

- Since the blocks of the same station are activated sequentially, the station processing time $t^{S}(N_k)$ is equal to the sum of its block processing times: $t^{S}(N_k) = \sum_{l=1}^{n_k} t^{b}(N_{kl}) + \tau^{S}$

Mixed integer program (MIP) for solving TLBP [4, 6].

Model notations:

q is the index of block, e.g., for the block l of a station k, $q = (k-1)n_0 + l$;

q_0 is the maximal possible value of q, $q_0 = m_0 n_0$;

$S(k) = \{(k-1)n_0 + 1, \ldots, kn_0\}$ is the set of block indices for a station k;

$Q(j)$ is the set of indices q (blocks) where operation j can be assigned;

$K(j)$ is the set of indices k (stations) where operation j can be assigned;

e is a set of operations which is an element of ES, $\overline{ES}$ or $\overline{EB}$;

$j(e)$ is some fixed operation from the set e.

$t_j = \lambda_j / s_j$ is the execution time of operation j if it is performed alone in a block.

$t_{ij} = \max\{\lambda_i, \lambda_j\} / \min\{s_i, s_j\}$ is the execution time of two operations i, j if they are performed in one block. In case s_j are all identical, this value is not used.

Variables:

X_{jq} is a binary decision variable (1 if operation j is assigned to block q and 0 otherwise);

$F_q \geq 0$ is an auxiliary variable for determining the time of block q;

$Y_q \in \{0, 1\}$ is an auxiliary variable that indicates if block q exists or not;

$Z_k \in \{0, 1\}$ is an auxiliary variable that indicates if station k exists or not. The variables Y_q and Z_k are used to count the number of blocks and stations, respectively.

To reduce the number of decision variables and constraints, the intervals of possible values of block $Q(j)$ and station $K(j)$ numbers for each operation are obtained from the analysis of all the constraints (for details, see [5]).

To reduce the number of decision variables and constraints, the intervals of possible values of block $Q(j)$ and station $K(j)$ numbers for each operation are obtained from the analysis of all the constraints (for details, see [5]).

The objective function is represented by (1). The precedence constraints are formulated in (2). Constraints (3) reflect the fact that each operation must be assigned to exactly one block. Constraints (4) determine the necessity of grouping certain operations in the same station. Constraints (5)–(6) deal with the impossibility of grouping certain operations in one block or executing certain operations at the same station, respectively. Constraints (7) and (7′) determine the block processing times: here condition (7) corresponds to the case of single operation in block, while (7′) covers the cases of two or more operations. In case s_j are all identical, inequality (7’) is redundant. Constraint (8) is the constraint on the cycle time. Constraints (9) ensure that block q exists in the design decision if and only if X_{jq} = 1 at least for one j. Constraints (10) ensure that station k exists in the design decision if and only if Y_q = 1 at least for one $q \in S(k)$. Constraints (11) guarantee that block q is created in station k only if block q – 1 exists for this station. Constraints (12) ensure that station k can be created only if station k – 1 exists at the line. Inequalities (11) and (12) mainly serve as symmetry-breaking cuts in this model (note that by a simple modification of (10) one could make these inequalities redundant). Bounds (13) are also imposed to reduce the polyhedron of linear relaxation.

$$\min W(Z_k, Y_q) = C_1 \sum_{k=1}^{m_0} Z_k + C_2 \sum_{q=1}^{q_0} Y_q \tag{1}$$

subject to:

$$\sum_{q \in Q(j)} q X_{jq} \geq \sum_{s \in Q(i)} s X_{is}, \quad (i, j) \in D; \tag{2}$$

$$\sum_{q \in Q(j)} X_{jq} = 1, \quad j \in N; \tag{3}$$

$$\sum_{j \in e \setminus \{j(e)\}} \sum_{q \in S(k) \cap Q(j)} X_{jq} = (|e| - 1) \sum_{q \in S(k)} X_{j(e)q}, \quad e \in ES,\ k \in K(j(e)) \tag{4}$$

$$\sum_{j \in e} X_{jq} \leq |e| - 1, \quad e \in \overline{EB},\ q \in \bigcap_{j \in e} Q(j); \tag{5}$$

$$\sum_{j \in e} \sum_{q \in S(k) \cap Q(j)} X_{jq} \leq |e| - 1, \quad e \in \overline{ES},\ k \in \bigcap_{j \in e} K(j); \tag{6}$$

$$F_q \geq (t_i + \tau^b)\, X_{iq}, \quad i \in \boldsymbol{N}, \quad q \in Q(i) \tag{7}$$

$$F_q \geq (t_{ij} + \tau^b)(X_{iq} + X_{jq} - 1), \quad i,j \in \boldsymbol{N}, \quad i < j, \quad q \in Q(i) \cap Q(j) \tag{7′}$$

$$\tau^S + \sum_{q \in S(k)} F_q \leq T_0, \quad k = 1,2,...,m_0; \tag{8}$$

$$Y_q \geq X_{jq}, \quad j \in \boldsymbol{N}, \quad q \in Q(j); \tag{9}$$

$$Z_k \geq Y_q, \quad k=1, 2,\ldots, m_0, \quad q = (k-1)n_0 + 1; \tag{10}$$

$$Y_{q-1} - Y_q \geq 0, \quad q \in S(k) \setminus \{(k-1)n_0 + 1\}, \quad k = 1, 2,\ldots, m_0; \tag{11}$$

$$Z_{k-1} - Z_k \geq 0, \quad k = 2, 3,\ldots, m_0; \tag{12}$$

$$F_q \in [0, T_0 - \tau^s - \tau^b], \quad q = 1, 2,\ldots, q_0. \tag{13}$$

References

[1] Baybars, I., 1986. A survey of exact algorithms for the simple assembly line balancing. Management Science 32, 909–932.

[2] Dolgui, A., Eremeev A, Guschinskaya, 2008. O. MIP-based GRASP and Genetic algorithm for balancing transfer lines. Submitted to Proc. Matheuristics Workshop, June 16-18, 2008, Bertinoro, Italy., 22p.

[3] Dolgui, A., Guschinsky, Levin, G., 2000. Approaches to balancing of transfer lines with blocks of parallel operations; Preprints No 8, Institute of Engineering Cybernetics of Academy of Sciences of Belarus/University of Technology of Troyes, 42p.

[4] Dolgui, A., Finel, B., Guschinsky, N., Levin, G. and Vernadat, F., 2006a. MIP approach to balancing transfer lines with blocks of parallel operations, IIE Transactions 38, 869–882.

[5] Dolgui, A., Guschinsky, N., Levin, G. , 2006b A special case of transfer lines balancing by graph approach, European Journal of Operational Research 168 (3), 732-746.

[6] Guschinskaya, O., Dolgui, A., 2006. A Comparative Evaluation of Exact and Heuristic Methods for Transfer Line Balancing Problem, in: Information Control Problems In Manufacturing 2006: A Proceedings volume from the 12th IFAC International Symposium, St Etienne, France, 17-19 May 2006, A. Dolgui, G. Morel, C. Pereira, eds., Elsevier Science, ISBN: 978-0-08-044654-7, Vol. 2, p. 395–400.

[7] Guschinskaya, O., Dolgui A., 2007. Balancing transfer lines with multiple-spindle machines using GRASP. Unpublished manuscript.