Facility Layout Problem with Buffer Space Allocation for Throughput and Material Handling Cost

Takashi IROHARA, Hideaki YAMASHITA and Yo ISHIZUKA

Abstract

We propose a new approach to optimize facility layout and buffer space allocation for production systems with a feed-forward configuration and variable processing times. Our objective is to efficiently find Pareto-optimal solutions for both throughput and material handling cost. We assume that work transfer is performed by conveyor belt or unlimited vehicles. Since the facility layout does not affect the throughput under these assumptions, we first calculate the throughput for every buffer space allocation using Markov analysis. Then the set of Pareto optimal facility layouts and buffer space allocations is searched using a genetic algorithm. Numerical examples illustrate the proposed approach.

Key words: facility layout, buffer space allocation, throughput, material handling cost, Pareto-optimal
1. Introduction

We herein propose a new approach to obtain an optimal combination of facility layout and buffer space allocation for job-shop production systems in which relatively large products are processed via variable routes in First-Come-First-Served (FCFS) manner and the processing time varies stochastically.

Thus far, the buffer space allocation problem and facility layout problem have been treated independently. Almost all manufacturing systems involve random elements such as variable inter-arrival times of orders, variable processing times, and occurrences of machine failure. This randomness causes the blocking and starving phenomena, and thus reduces manufacturing efficiency. For such systems, it is a common practice to place some buffer spaces in front of the machines so as to absorb the fluctuation and improve the manufacturing efficiency of the system. In general, when the capacities of any of the buffers are increased, the throughput of the system increases with a decreasing rate. Buffer allocation problems consider the situation in which the total amount of available buffer space is limited, and specify how many buffer spaces should be allocated to each machine in order to attain the maximum throughput. Although a number of studies have been devoted to the buffer space allocation problem (see [9], [18]), the layouts of these buffer spaces and their influence upon material handling cost have not yet been considered. However, for cases in which the physical size of the product or work in process is relatively large compared with the machine area itself, the buffer space cannot be ignored and affects the material handling cost. In other words, there is a trade-off between production efficiency and material handling cost. Thus, it is natural to consider the optimal combination of the layout and the buffer space allocation simultaneously in order to attain the best manufacturing efficiency.

On the other hand, most studies on the facility layout problem in production systems target deterministic systems. Koopmans and Beckman [11] introduced the quadratic assignment problem (QAP) to model the problem of locating interacting departments of equal area. This objective is to minimize the so-called “material handling cost,” which is essentially the sum of the material flow multiplied by the travel distances, and this layout model has become the most basic. After this research, numerous papers on facility layout problems have been published (see [12] and [15] for extensive reviews).

Recent studies on facility layout problems extend the basic model in several ways. A number of new layout expressions that can handle unequal department area and variable shapes have been proposed (see [1], [14], [19]). The objective functions of these “new” studies on facility layout problems remain unchanged from the Koopmans and Beckman QAP model and are still the material handling cost. Using the material handling cost as a measure of the performance of layouts is justified based on the following observation: “Between 20 to 50% of the total operating expenses within manufacturing are attributed to materials handling,” and an efficient layout can “reduce these costs by at least 10 to 30%” [17].

However, if the goal of the facility layout problem is to improve manufacturing efficiency, it is appropriate to use manufacturing efficiency as the direct measure of performance for the layouts. Recently, a number of authors have suggested the need for considering other objective functions for facility layout problems. Hamamoto et al. [8] developed a simulation-based model. The objective is to maximize the throughput rate and minimize the conveyance time per trip, and the solution method is based on a genetic algorithm. Comparisons indicate that their proposed method significantly outperforms traditional layout methods for minimizing conveyance time per trip under the same throughput rate. Azadivar and Wang [2] pointed out that in today’s short-cycle time production environments, the timing of materials movement may have a greater impact on the productivity of the system than its cost. Their optimization model takes into consideration the dynamic characteristics and operational constraints of a system as a whole. The performance measures are cycle time and productivity. Genetic algorithms are used to optimize the layout for manufacturing effectiveness, while simulation serves as a system performance evaluation tool. A recent study by
Castillo and Peters [4] formulates a problem to find the optimal unit load size (batch size) and facility layout (machine allocation to prefixed locations) that achieve minimal work-in-process (WIP) under variable processing times. Their optimization method relies on simulated annealing, and the evaluation technique for production efficiency (work-in-process) is based on an approximation scheme for a number of queuing systems. Benjaafar [3] presents a model that captures the relationship between facility layout and congestion-related measures of performance. The objective is to minimize WIP. He showed that layouts obtained using a WIP-based formulation can be very different from those obtained using conventional QAP formulations. He assumed that there is always sufficient capacity for queueing. However, as he also suggests in his paper, there are manufacturing environments where the capacity is limited. Limits in queue sizes can lead to the occasional blocking and starvation of the processing departments and the material handling system, and the analysis of such systems is considerably more complex.

Ishizuka et al. [10] constructed a new model with limited queue capacity and proposed a problem to find an optimal combination of facility layout and buffer space allocation that attains maximal throughput. Their model is slightly different from the present paper, in which the order and route of each part are fixed in advance, and hence there is no need for any Markov-type analysis to evaluate production efficiency. All of these studies considered, to some degree, random elements such as manufacturing efficiency as performance measures, and integrated optimization of layouts and other structural and/or operational design.

In the present study, we consider a multi-objective facility layout problem that efficiently finds Pareto-optimal solutions for both throughput and material handling cost.

The basic assumptions of the proposed model are:
1) Each department consists of a machine and its buffer space;
2) The area of the buffer space cannot be ignored (the size of each department is the sum of the sizes of its machine and buffer spaces);
3) Departments are connected arbitrarily in a feed-forward manner (without a closed loop), a completed job at a department is transferred to another department (or leaves the system) with a fixed probability determined by the from-to chart, and at each department, jobs are processed in the FCFS manner;
4) The processing time distribution at each machine is exponential for computational tractability;
5) The job transfer is performed by high-speed conveyor belts or unlimited vehicles, thus the jobs do not need to wait for transportation between departments, so the conveyance time does not affect the throughput;
6) The blocking mechanism is blocking-before-processing where the machine may not start processing a new item until space becomes available in the next buffer;
7) The performance of the system is measured by the material handling cost and production efficiency (throughput or lead time); and
8) The from-to chart does not change during a planning horizon.

When the flow of materials between departments changes during horizon, the static facility layout problem (SFLP) becomes dynamic, and this problem is known as the dynamic facility layout problem (DFLP) [13], which minimizes the sum of material handling cost for all periods and the sum of the rearrangement costs. The DFLP is based on the assumption that changes in flow can occur in the future. The flow data for each period are forecasted and the rearrangement cost is also estimated. When we solve this DFLP, SFLP is a part of the problem and it is essential to optimize SFLP efficiently. Therefore, our proposed approach would be very important not only for SFLP, but also for DFLP.

Since the facility layout does not affect throughput under these assumptions, we can calculate the throughput (or lead time) for every buffer space allocation using Markov analysis. It may be possible to find Pareto-optimal solutions for both throughput and material handling cost in two-phases; that is, in the first phase, one solves the buffer allocation problem to attain the maximum throughput given the total amount of buffer spaces, and then obtains the facility layout which minimizes
the sum of material handling cost for every optimal buffer allocation in the second phase. However, this method may miss some Pareto-optimal solutions since some of the non-optimal buffer allocations may realize small material handling costs and result in Pareto-optimal solutions. The risk becomes large as the material handling costs deeply depend on the buffer allocations. Consequently, we simultaneously search the set of Pareto-optimal facility layouts and buffer space allocations using a genetic algorithm (GA). In the gene representation, the layout of the department and buffer space allocation are coded in one chromosome in order to incorporate the influence of buffer space allocation on the material handling cost. The layout expression is based on the concept of the Flexible Bay Structure (FBS), which can only produce rectangular departments. In this paper, a GA-based algorithm was proposed, however, Simulated Annealing, Tabu Search, Ant Colony optimization and many other kinds of heuristics would be applicable. Numerical examples illustrate the proposed approach.

2. Formulation

The following are definitions and symbols for the input parameters:

- \( M \): Number of departments;
- \( S \): Set of departments (\( \{1, 2, \ldots, M\} \));
- \( B_{\text{total}} \): Total amount of buffer spaces;
- \( a_i \): Minimum (default) area needed for machines and workers at department \( i \);
- \( u \): Area of one buffer space;
- \( r_i^{\text{max}} \): Maximum aspect ratio of department \( i \);
- \( R^{\text{max}} \): Maximum aspect ratio of entire building;
- \( \mu_i \): Process (service) rate of the machine at department \( i \); and
- \( f_{i,j} \): Flow volume from department \( i \) to department \( j \) (element of the from-to chart).

To formulate our problem, we introduce the following variables:

- \( L \): Vector indicating the facility layout;
- \( B_i \): Number of buffer spaces allocated to department \( i \) (including machine \( i \) itself);
- \( B \): Buffer allocation vector (= \( (B_1, \ldots, B_M) \));
- \( A_i(B_i) \): Area of department \( i \) (= \( a_i + (B_i-1)u \)); and
- \( \text{dist}_{i,j}(L, B) \): Rectilinear distance between centers of departments \( i \) and \( j \).

Let

\[
MHC(L, B) = \sum_{i,j} f_{i,j} \cdot \text{dist}_{i,j}(L, B)
\]

be the material handling cost (MHC) and \( TH(B) \) the steady-state throughput of the system. Then, our problem, which is to find an optimal layout for the departments, and an optimal buffer allocation for the departments is formulated as the following multi-objective facility layout problem:

\[
\begin{align*}
(P) \quad & \min_{L,B} \left( MHC(L, B) \right) \\
& \text{subject to } (L, B) \in \mathcal{L} \times \mathcal{B},
\end{align*}
\]

where \( \mathcal{L} \) is a set of feasible layouts, and \( \mathcal{B} \) is a set of feasible buffer space allocations. A concrete definition of \( \mathcal{L} \) depends on the representation of the layouts and should be defined by some constraints such that no two department areas overlap, the aspect ratio of each department must be less than or equal to the allowable value \( r_i^{\text{max}} \), and so on. We assume that the constraint set \( \mathcal{B} \) is defined as:

\[
\mathcal{B} = \left\{ B \in Z^M \mid \sum_{i=1}^M B_i \leq B_{\text{total}}, \quad B_i \geq 1, \quad i=1, \ldots, M \right\}
\]

where \( Z^M \) denotes the set of \( M \)-dimensional integer vectors. Note that, in general, the buffer space allocation \( B \) and facility layout \( L \) cannot be optimized independently. There is a tradeoff between MHC and \( TH \); increasing the total amount of the buffer spaces reduces the reciprocal of the throughput \( 1/TH \) but enlarges the areas \( A_i(B_i) \) of the departments and the distances \( \text{dist}_{i,j}(L, B) \) between departments, resulting in a higher material handling cost MHC.

Since problem (P) is a usual (multi-objective) combinatorial optimization problem, we can solve it using a GA provided that the values of objective functions MHC and TH can be evaluated easily.

3. GA-based Solution Method

Genetic algorithms are heuristic search and optimization techniques that imitate the natural selection and biological evolutionary processes [7]. A generation consisting of surviving indi-
viduals from the previous population and new offspring is generated through reproduction by means of crossover, mutation, and selection of parental chromosomes. The only requirement for applying a GA is that the values of objective functions can be optimized at any decision variable. In this section, we first describe our approach to calculating throughput \( TH \) via the decomposition method, and we then present a solution method based on a GA.

### 3.1 Calculation of the Throughput \( TH \)

For the sake of simplicity, we assume that Department 1 is the only department that accepts raw materials from outside. In addition, we assume that the branching probabilities \( r_{i,j} \) are determined from the from-to chart as follows:

\[
r_{i,j} = \frac{f_{i,j}}{\sum f_{i,j}^b}.
\]

Let \( d_i \) (\( i = 1, \ldots, n-1 \), \( d_i = i+1, \ldots, n+1 \)) be the destination department of the item that departs next from department \( i \). If department \( i \) is the last item for the set, we set \( d_i = n+1 \). Let \( m_i \) (\( i = 2, \ldots, n, m_0 = 0, \ldots, B_i \)) be the number of items in department \( i \), including that being processed by machine \( i \). Define \( d = (d_1, \ldots, d_{n-1}) \) and \( m = (m_2, \ldots, m_n) \). Then, state \( (d, m) \) has the Markov property. Denoting the steady-state probability of the state by \( \pi(d, m) \), \( \pi(d, m) \) can be obtained by solving the following set of equations:

\[
\sum_{i=1}^{n} \sum_{j=1}^{m} \pi(d, m) = 1
\]

(1)

and

\[
\sum_{i=1}^{d} \sum_{m} \pi(d, m) = 1
\]

(2)

where \( m_i > 0 \), \( B_{n+1} = \infty \). \( d_i, d_{i+1}, \ldots, d_{n-1}, e_i \) is an \((n-1)\)-dimensional unit vector of which the \( i \) th element is equal to 1, and \( \{ \cdot \} \) is the indicator function. If \( \pi(d, m) \) is obtained, then the throughput can be expressed by:

\[
TH(B) = \sum_{i=1}^{n} \{1[d_i = n+1]1[m_i > 0]m_i \pi(d, m).
\]

However, the number of unknowns of Eqs. (1) and (2) increases as the number of departments and/or the number of total buffer spaces increases, and then the set of equations becomes intractable. Therefore, some approximations are necessary in order to obtain the steady-state probability \( \pi(d, m) \) for large systems. In the present paper, we use the decomposition method; that is, the system is decomposed to each department, and each department is analyzed in isolation. In the analysis of one department, the state information of other departments is incorporated to the transition rate of the steady-state equations for the department.

Let \( \pi_i(d, m) \) be the steady-state probability of state \((d, m)\). Although \((d, m)\) does not have the Markov property, we assume that \( \pi_i(d, m) \) satisfies the steady-state equations:

\[
\lambda_i \pi_i(d, m) = \sum_{j=1}^{m} \pi_{i,j}(d, m) + \sum_{j=1}^{m} \sum_{l=i}^{n} \lambda_{j,l} \mu_{i,j} \pi_{j,l}(d, m),
\]

(3)

where \( \lambda_i \) is the input rate of items to department \( i \) given that department \( i \) is not full and \( \mu_{i,j} \) is the output rate of items from department \( i \) given that the destination department of the processing item is \( j \), and they are given by:

\[
\begin{align*}
\lambda_i &= \mu_i r_{i,i} + \sum_{j=1}^{m} \pi_{i,j}(d_i, m_i) \\
\mu_{i,j} &= (1 - \sum_{k=1}^{m} \pi(k, B_i)) (j = i + 1, \ldots, n),
\end{align*}
\]

Note that \( \lambda_i \) and \( \mu_{i,j} \) depend on the state of departments to which department \( i \) is connected. We can calculate (approximate values of) \( \pi(d, m) \) and \( TH(B) \) using the following algorithm, where \( s \) is the iteration number.

**Step 1.** Initialize \( \pi(d, m)^{(0)} \) for \( i = 2, \ldots, n \), and set \( s = 0 \).

**Step 2.** For \( i = 2, \ldots, n, j = i + 1, \ldots, n \), set

\[
\begin{align*}
\lambda_i &= \mu_{i,i} r_{i,i} + \sum_{j=1}^{m} \pi_{i,j}(d_i, m_i)^{(s)} \\
\mu_{i,j} &= (1 - \sum_{k=1}^{m} \pi(k, B_i)^{(s)}).
\end{align*}
\]
We then obtain \( \pi_i(d_i, m_i)^{(s)} \) by solving Eq. (4).

**Step 3.** Calculate the throughput by:

\[
TH(B)^{(s)} = \sum_{i=1}^{n} \sum_{j=1}^{m_i} \mu_i \pi_i(n+1, m_i)^{(s)}.
\]

If \( TH(B)^{(s)} - TH(B)^{(s-1)} < \varepsilon \), then stop; otherwise set \( s := s + 1 \) and go to Step 2.

### 3.2 String Representation of Layout and Buffer Space

In the proposed model, the area of each department depends on the number of buffer spaces allocated, because each department consists of not only a machine, but also a number of buffer spaces. This means that the department area is not fixed during their optimization process. This characteristic is quite different from previous facility layout problems, because in most previous facility layout models \([15]\) the area of each department is input data and the buffer space for each department is not considered. The use of the Flexible Bay Structure (FBS) \([16]\) makes it possible for the present problem to find the only feasible solution; that is, rectangular departments with no overlap, despite the changeable department area.

The decision variables of the proposed model are the layout vector \( L \) and the buffer allocation vector \( B \). Then, string representation requires expression of the location of the department and the number of buffer spaces allocated. Thus, the genetic string consists of two parts for vector \( L \) and \( B \).

FBS is applied to the expression of layout vector \( L \). The prescribed rectangular area is divided in one direction into bays of varying width. Each bay is then divided into rectangular departments of equal width but different lengths. The bays are flexible in that their widths vary with their number and contents. In the expression of FBS, layout vector \( L \) consists of a sequence of the departments and breakpoints for the bays, as shown in Fig. 1. This sequence represents the order of departments, bay by bay, read from top to bottom and left to right. The breakpoints indicate where the breaks between bays occur in the sequence.

String representation of buffer space allocation vector \( B \) is the sequence of volume of the buffer allocated to each department. In the example in Fig. 1, \( B=(01002) \) means that one buffer space is allocated to Department 2, and two buffer spaces are allocated to Department 5.

### 3.3 Genetic Operations

After a population of strings is initialized, genetic strings are evaluated based on a fitness function. Subsequently, a new population of strings for the next generation is generated by crossover and mutation.

The crossover operator selects two parent chromosomes and generates two offspring chromosomes with the specified probability of crossover. We simply use the vector \( (L, B) \) as the string, and apply a GA with a partially matched crossover (PMX) \([5], [7]\), for the sequence of departments and Affine Crossover \([6]\) for the breakpoints and buffer allocation vector. PMX is done by swapping a number of genes between a pair of selected strings at a randomly selected position. This crossover is appropriate for exchanging the gene while respecting the absolute gene position. That's why we applied PMX to search for department sequence. However, in the case of breakpoints and buffer allocation vector, only exchanging the position of the gene cannot explore the new search space. Affine Crossover is an arithmetical crossover and it can produce a variety of new gene combinations. We use the mutation only for the sequence of departments. This is done by selecting two departments randomly and swapping their positions in the child string. In these processes, only feasible solutions \( (L \in \mathcal{L}(B), B \in \mathcal{B}) \) are accepted. After crossover and mutation, the fitness value is calculated.

The task of selection in the GA is to allocate the reproductive opportunities to each chromosome such that chromosomes with higher fitness are more likely to survive to the next generation. In the proposed algorithm, we first preserve the Pareto-optimal solution from the pop-
ulation in a new generation, and then parallel selection is applied to the remaining population in order to satisfy the population number of the next generation. The method is such that two parents are selected by a roulette wheel, which is a mechanism for selecting individuals based on their fitness. The higher the fitness, the greater the probability that the individual will be selected. Hence, a suitable fitness function must increase as the corresponding cost function decreases.

We define the fitness functions for the material handling cost and the throughput as $1/MHC(L, B)$ and $TH(B)$, respectively.

4. Numerical Example

As an illustrative example, we consider a 32-department system with the configuration parameters described in Table 1. Here, the default size of each department $a_i \in \{1, 2, 3, 4\}$ and flow volume $f_{i,j} \in \{0, 1, 2, 3, 4\}$ were generated at random, as shown in Tables 2 and 3, respectively.

Characteristic points to be noted in this example are:

- Size of buffer space is relatively large ($u=3$) compared to default department sizes ($a_i$);
- Aspect ratio of each department must not exceed 2.0; and
- There is no constraint on the aspect ratio of the entire building ($R_{\text{max}}=\infty$).

We considered the following problem:

$$\min_{L,B} \left( \frac{1}{MHC(L, B)} \right)$$

subject to $(L, B) \in \mathcal{L} \times \mathcal{B}$,

where $TH(B)$ is the approximated throughput.

Table 1 Configuration parameters of the example

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of departments $(M)$</td>
<td>32</td>
</tr>
<tr>
<td>Available total buffer spaces</td>
<td>32</td>
</tr>
<tr>
<td>Default sizes of departments $(a_i)$</td>
<td>See Table 2</td>
</tr>
<tr>
<td>Buffer sizes $(u)$</td>
<td>3</td>
</tr>
<tr>
<td>Maximum aspect ratios of departments $(r_{\text{max}})$</td>
<td>2.0, $i=1, \ldots, 32$</td>
</tr>
<tr>
<td>Maximum aspect ratio of entire building $(R_{\text{max}})$</td>
<td>$\infty$</td>
</tr>
<tr>
<td>Process rates $(\mu_i)$</td>
<td>0.1, $i=1, \ldots, 32$</td>
</tr>
<tr>
<td>Flow volumes $(f_{i,j})$</td>
<td>See from-to chart (Table 3)</td>
</tr>
</tbody>
</table>

Table 2 Default areas of departments

<table>
<thead>
<tr>
<th>Department</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
<th>30</th>
<th>31</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_i$</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 3 From-to chart

```
1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 Out
1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 Out
```
obtained by the method described in Section 3.1 and $L$ is defined by the FBS representation and aspect ratio constraints. For this problem, we applied the GA to obtain the set of (approximate) Pareto solutions.

We recorded the set of Pareto solutions in each generation. Figure 2 shows these Pareto solutions at generations 1, 40 and 200. The layouts and the buffer space allocations corresponding to solutions A~F in Fig. 2 are depicted in Figs. 3~8. In Figs. 3~8, the shadowed area represents the size of the buffer spaces allocated to the department (not including the machine size).

Clearly, initial solutions A and B are far from the true Pareto solution set. Solutions C~F are considered to be near the true Pareto solution set. Solution C is the minimal material handling cost solution, Solution D is the maximal throughput solution, and Solutions E and F are “interim” solutions. Note that some buffer spaces are allocated in the minimal handling
cost solution (Solution C). Of course, fewer total buffer spaces generally results in a lower material handling cost. In this case, however, if we reduce the number of buffer spaces, the variance in department size becomes large and there could be no feasible layout that satisfies the aspect ratio constraints. Thus, as far as the aspect ratio constraints exist, buffer spaces work to both increase throughput and smooth the sizes of the departments. In minimal material handling cost solutions (Solutions C and E), the small departments that have no buffer space tend to be located inside the building so as to reduce travel time.

5. Conclusion

We have formulated a multi-objective facility layout problem for job-shop production systems, and using an approximate Markov analysis and a genetic algorithm, we have developed a technique for generating Pareto-optimal solutions with respect to the material handling cost and throughput. After generating the Pareto-optimal solutions (combinations of facility layouts and buffer space allocations), the decision-maker can choose his/her preferred solution from among them.

Although the model presented herein is rather
simple, our approach can be applied to more realistic models. For example, we can consider the case in which the transfer times cannot be ignored. For such a case, the transfer time distribution of each origin-destination pair is approximated by a phase-type distribution so that similar approximate Markov analysis (decomposition method) is applicable. Furthermore, if there is a common buffer space (warehouse) that is a temporary buffer for jobs having destination department buffers that are full, we can determine the optimal facility (including the common buffer space) layouts and optimal buffer space allocations by an approach similar to that described herein. These extensions will be investigated in future research.

References


