IIE Transactions
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/uiie20

Design of distributed layouts

MAHER LAHMAR [a] & SAIF BENJAAFAR [b]
[a] Department of Industrial Engineering, University of Houston, Houston, TX, 77204-4008, USA
[b] Graduate Program in Industrial Engineering, Department of Mechanical Engineering, University of Minnesota, Minneapolis, MN, 55455-0111, USA

Available online: 23 Feb 2007

To cite this article: MAHER LAHMAR & SAIF BENJAAFAR (2005): Design of distributed layouts, IIE Transactions, 37:4, 303-318

To link to this article: http://dx.doi.org/10.1080/07408170590517015

IIE Transactions (2005) **37**, 303–318
Copyright © "IIE"
ISSN: 0740-817X print / 1545-8830 online
DOI: 10.1080/07408170590517015

Design of distributed layouts

MAHER LAHMAR[1] and SAIF BENJAAFAR[2]

[1]*Department of Industrial Engineering, University of Houston, Houston, TX 77204-4008, USA*
E-mail: mlahmar@uh.edu
[2]*Graduate Program in Industrial Engineering, Department of Mechanical Engineering, University of Minnesota, Minneapolis, MN 55455-0111, USA*
E-mail: saif@ie.umn.edu

Received October 2003 and accepted July 2004

Distributed layouts are layouts where multiple copies of the same department type may exist and may be placed in non-adjoining locations. In this paper, we present a procedure for the design of distributed layouts in settings with multiple periods where product demand and product mix may vary from period to period and where a relayout may be undertaken at the beginning of each period. Our objective is to design layouts for each period that balance relayout costs between periods with material flow efficiency in each period. We present a multi-period model for jointly determining layout and flow allocation and offer exact and heuristic solution procedures. We use our solution procedures to examine the value of distributed layouts for varying assumptions about system parameters and to draw several managerial insights. In particular, we show that distributed layouts are most valuable when demand variability is high or product variety is low. We also show that department duplication (e.g., through the disaggregation of existing functional departments) exhibits strong diminishing returns, with most of the benefits of a fully distributed layout realized with relatively few duplicates of each department type.

1. Introduction

We consider the design of plant layouts in environments with multiple products and a planning horizon of several periods where demand volumes and product mix vary from period to period. In these environments, there is a need to design layouts that are either easily reconfigurable or robust enough so that they offer acceptable performance under most likely product demand scenarios. For reconfigurability, plants may adopt a *dynamic* layout that can change from period to period. In this case, the challenge is to design a layout that balances relayout costs between periods with material flow efficiency within each period. For robustness, plants may adopt a single layout that balances the material flow requirement needs of all future periods (this can, of course, be viewed as an instance of a dynamic layout where relayout costs are prohibitively expensive). In this case, the challenge is to ensure that a selected layout guarantees an acceptable degree of efficiency in each period.

In practice, robustness tends to be more popular than reconfigurability. Manufacturing firms are reluctant to incur the disruption to production that is usually associated with relayout. Consequently, firms attempt to adopt layouts that are sufficiently flexible to accommodate a wide range of production requirements. In most cases, this translates into functional layouts where resources of the same type are grouped into functional departments and placed in adjoining locations. The placement of these departments relative to one another is determined by an aggregate measure of material flow cost over all future periods within the planning horizon.

A functional organization of the plant has the benefit of limiting the commitment of the firm to a particular flow pattern and offers some economies of scale in operating these departments. More significantly, it provides for capacity pooling for each resource type and effective load allocation among duplicates of the same resource type. Unfortunately, a functional layout is also notorious for its material flow inefficiencies and scheduling complexity (Benjaafar *et al.*, 2002). Since a functional layout is not optimized with a particular product in mind, material flow tends to be inefficient for most products. This is particularly the case when product variety is high and demand is variable; an effect that is compounded when functional departments are large and consist in several individual resources.

An alternative to a functional layout is a cellular layout, where resources are partitioned into cells, each devoted to a family of products that share similar processing requirements. Although a cellular layout simplifies workflow and reduces material handling effort (Heragu, 1994), it is effective only when product families are sufficiently stable and production volumes are relatively large. Otherwise, frequent cell redesigns would be required (Irani *et al.*, 1993; Montreuil *et al.*, 1999) or significant intercell flows must be

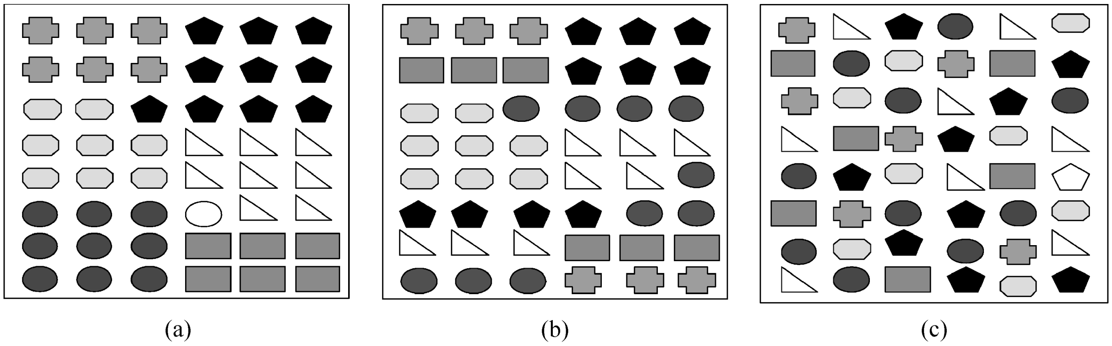

Fig. 1. Layouts with varying degrees of distribution: (a) functional layout; (b) partially distributed layout; and (c) maximally distributed layout.

allowed. To minimize intercell flows, resources are often duplicated, leading to higher investment costs and unbalances in utilization among resource duplicates.

To mitigate the limitations of cellular layouts, hybrid layouts that combine the features of cellular and functional layouts have been suggested, such as overlapping cells (Irani *et al.*, 1993), cells with machine sharing (Benjaafar, 1995), virtual cells (Drolet, 1989), fractal cells (Montreuil *et al.*, 1999), and cells with a remainder functional cell (Irani and Huang, 2000). In each case there is an attempt to balance the benefits of sharing resources among multiple products (i.e., improving resource utilization and minimizing investment costs) but ensuring efficient material flows (i.e., reducing material handling costs). In each case, there is also an attempt to increase the flexibility and robustness of traditional cells. Although an improvement, many of these alternatives remain bounded by their cellular structure (see Benjaafar *et al.* (2002) for further discussion).

In this article, we consider *distributed* layouts as an alternative to functional layouts that does not presume a cellular structure. In a distributed layout, functional departments are disaggregated into smaller subdepartments. These subdepartments are then distributed strategically throughout the plant floor. Intuitively, this disaggregation and distribution should allow the plant to hedge against future changes in product mix, product routings and demand volumes. The distribution of similar resources in different areas of the plant could reduce the distances between department types and could increase the accessibility to all departments from different regions of the layout. As a result, efficient flows could be more easily found for a larger set of product routings, which would then tend to diminish the need for rearranging the layout even when production requirements change significantly. Although a distributed layout does not presuppose a cellular structure, it can serve as the basis for one. For example, based on demand realization, temporary cells, consisting of adjoining subdepartments could be dedicated to particular product lines or job orders. These cells would be disbanded once the products are phased out or once customer orders are completed. The individual replicates would then be free to participate in new cells (Drolet, 1989). Example layouts with varying degrees of department disaggregation and distribution are shown in Fig. 1. A distributed layout with virtual cells is illustrated in Fig. 2.

The concept of distributed layouts is not new. Montreuil *et al.* (1991) suggest a maximally distributed layout, or *holographic*, layout where functional departments are fully disaggregated into individual machines which are then placed as far from each other as possible to maximize distribution. Benjaafar and Sheikhzadeh (2000) develop a procedure for designing robust distributed layouts where they take into account demand distribution information to determine department placement and flow allocation. Deterministic versions of the problem are discussed by Urban *et al.* (2000). Drolet (1989) illustrates how a distributed layout configuration could be used to form virtual cells that are temporarily dedicated to a job order. Askin *et al.* (1997) compare the operational performance of a distributed layout with cellular, functional and hybrid layouts using simulation.

In this article, we consider the design of distributed layouts in a multiple period setting, where product demand and product mix vary from period to period. We allow for the possibility of a relayout at the end of each period. Our objective is to design layouts for each period that balance relayout costs between periods with material flow efficiency within each period. We present a multi-period model for

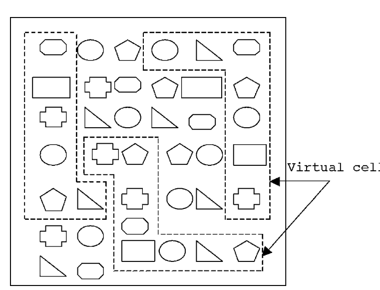

Fig. 2. Using a distributed layout to form virtual cells.

jointly determining layout and flow allocation and offer a decomposition-based solution procedure. We use our solution procedure to examine the impact of department disaggregation and distribution on total cost under varying levels of material handling and relayout costs, number of planning periods, product variety, and demand variability. We find that distributed layouts are most valuable when demand variability is high or product variety is low. We also find that department duplication (e.g., through the disaggregation of existing functional departments) exhibits strongly diminishing returns, with most of the benefits of a fully distributed layout realized with relatively few duplicates of each department type. This suggests that rarely would full department duplication and distribution be justified.

Our work is related to two streams in the layout literature: (i) dynamic facility layout; and (ii) facility layout under uncertainty. The dynamic facility layout problem arises in settings with multiple periods where it is possible to reconfigure the layout at the beginning of each period. Assuming demand information for each period is available at the initial design stage, the objective is to identify a layout for each period such that both the material handling and relayout costs are minimized over the planning horizon. Rosenblatt (1986) was the first to formulate a version of the dynamic facility layout problem. He presents a model and a solution procedure for determining optimal layouts for each period over a specified planning horizon. Improvements to Rosenblatt's procedure are offered by Batta (1987), Palekar (1992), Urban (1992, 1998) and Balakrishnan (1993). Heuristic approaches to the dynamic facility layout problem are discussed in Urban (1993), Kaku and Mazzola (1997) and Kochhar and Heragu (1999). A review of literature on the dynamic facility layout problem can be found in Balakrishnan and Cheng (1998).

The modeling of uncertainty in layout design was first discussed by Shore and Tompkins (1980). They consider a single-period problem with a discrete set of production scenarios, each with a certain probability, and use expected material handling cost as the layout design criterion. Rosenblatt and Kropp (1992) show that this problem reduces to a deterministic layout problem with aggregate flows that are a weighted average of the scenario-specific ones. Rosenblatt and Lee (1987) consider the case where the probabilities of different possible scenarios are difficult to know. They discuss a robustness-based approach where the design criterion is to minimize the difference between the selected layout and the optimal layout under each scenario. Palekar *et al.* (1992) consider a multi-period version of the problem where the transition from one period to another is governed by a probability transition matrix. They solve the problem using dynamic programming for small-sized problems and heuristics for larger ones. Kouvelis and Kiran (1991) also consider a multi-period problem in which they include a constraint on throughput rate which they evaluate using a closed queueing network model. Other variations on the multi-period problems with discrete demand scenarios are considered by Montreuil and Laforge (1992), Kouvelis *et al.* (1992) and Yang and Peters (1998), among others. The effect of uncertainty on the operational performance of a layout, as measured primarily by congestion, is discussed in Benjaafar (2002) and the references therein. Comprehensive reviews of the layout literature can be found in Meller and Gau (1996), Balakrishnan and Cheng (1998) and Benjaafar *et al.* (2002).

The models we present in this paper are envisioned to be useful in settings where demand in each period can be characterized reasonably well. Therefore, they are most applicable when planning periods are relatively short (say weeks or months) and period to period variability is mostly due to anticipated changes in customer orders, demand seasonality, scheduled launch of new products, initiation or completion of supply contracts, or other planned changes in the production profile. A good example is contract manufacturing in the electronics industry. As noted in Benjaafar *et al.* (2002), large contract manufacturers, such as Solectron or Flextronics, tend to hold supply contracts from several original equipment manufacturers for a wide range of products. Given the rapid changes of technology, the typical length of these contracts is months rather than years. Hence, over a period of 6 months to a year, production demand is relatively well known, but it can vary significantly from month to month based on the start and expiration dates of various contracts.

Electronics manufacturing also offers a good example of an industry where department duplication and distribution could be implemented. Automated chip placement machines, inspection workstations, and packaging lines, are example departments that are typically present in multiple copies within the same plant and can be placed in different areas of the plant without significant loss of economies of scale (see Benjaafar *et al.* (2002) and the references therein for additional examples and discussion).

The remainder of this article is organized as follows. In Section 2, we provide a formulation of the dynamic distributed layout problem. In Section 3, we develop upper and lower bounds on the objective function. In Section 4 we present exact and heuristic solution procedures. In Section 5, we present numerical results and derive several managerial insights. In Section 6, we offer concluding comments.

2. Problem formulation

In this section, we formulate the dynamic distributed layout problem. Our formulation shares a similar structure to the classic dynamic layout problem. However, there are important differences. We allow for the possibility of multiple copies of the same department and for these copies to be placed independently of each other. Consequently, in addition to determining the department location at each period, we must determine the flow allocation among these departments. This means that we have a multi-period joint layout and flow allocation problem. In order to carry out the flow allocation, we must explicitly model the routing of each product, its processing requirement at each department, and its demand for each period. We must also explicitly model the production capacity of each department, which can vary from copy to copy and from period to period.

Our objective is to design a layout that minimizes the sum of material flow costs and rearrangement costs over a planning horizon consisting of T periods. Production requirements (i.e., number of products, demand for each product, process sequences and processing times, and department capacities) are assumed known for each period. In keeping with standard layout design formulation, we assume that material flow costs are linearly related to distance traveled. We use the following notation:

$$x_{nikt} = \begin{cases} 1 & \text{if } n^{\text{th}} \text{ department duplicate of type } i \text{ is assigned location } k \text{ at period } t, \\ 0 & \text{otherwise;} \end{cases}$$

- v_{nimjpt} = volume of flow due to product p between n^{th} duplicate of department i and m^{th} duplicate of department j at period t;
- v_{ijpt} = total flow volume due to product p between department duplicates of type i and department duplicates of type j at period t;
- d_{kl} = travel distance between location k and location l;
- t_{nip} = processing time per unit load of product type p at department duplicate n of type i;
- c_{kl} = cost of moving a unit load from department k to department l;
- C_{ni} = capacity (available operation time) of department duplicate n of type i;
- r_{nikl} = cost of rearranging department duplicate n of type i from location k to l;
- T = total number of periods;
- N_i = total number of department duplicates of type i;
- N = total number of department types;
- M = total number of locations; and
- P = total number of product types.

The layout design and flow allocation problem can now be formulated as follows:

(P) Min z

$$= \sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} v_{nimjpt}x_{nikt}x_{mjlt}c_{kl}d_{kl} + \sum_{t=1}^{T-1}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{k=1}^{M}\sum_{l=1}^{M} x_{nikt}x_{nil(t+1)}r_{nikl}, \quad (1)$$

subject to

$$\sum_{i=1}^{N}\sum_{n=1}^{N_i} x_{nikt} = 1 \qquad \forall k, t, \quad (2)$$

$$\sum_{k=1}^{M} x_{nikt} = 1 \qquad \forall i, n, t, \quad (3)$$

$$\sum_{m=1}^{N_j}\sum_{n=1}^{N_i} v_{nimjpt} = v_{ijpt} \qquad \forall i, j, p, t, \quad (4)$$

$$\sum_{i=0}^{N}\sum_{n=1}^{N_i} v_{nimjpt} = \sum_{q=0}^{N}\sum_{r=1}^{N_q} v_{mjrqpt} \qquad \forall j, m, p, t, \quad (5)$$

$$\sum_{p=1}^{P}\sum_{i=0}^{N}\sum_{n=1}^{N_i} v_{nimjpt}t_{mjp} \le C_{mj} \qquad \forall j, m, t, \quad (6)$$

$$x_{nikt} = 0, 1 \qquad \forall i, n, k, t, \quad (7)$$

$$v_{nimjpt} \ge 0 \qquad \forall i, n, j, m, p, t. \quad (8)$$

The above model treats both department locations (x_{mjkt}) and volume of flow between individual departments in each period (v_{nimjpt}) as decision variables, with the objective of minimizing the sum of material handling costs and relayout costs over all periods. The constraints of Equations (2) and (3) ensure that each department duplicate is assigned to one location and each location is assigned to one department duplicate. When the number of locations exceeds the number of department duplicates, dummy departments with zero flows and zero rearrangement costs may be used, without loss of optimality, to account for the difference. The constraints of Equation (4) equate the amount of flow between multiple duplicates of departments of types i and j to the amount of flow between departments of type i and departments of type j. The constraints of Equation (5) ensure that the amount of input and output flow (per product) to and from a department are equal. Note that we added a dummy department 0 with a single copy ($N_0 = 1$) that serves as an entry/exit department. Its main purpose is to balance the flow equation at the beginning and the end of the sequence and is not considered in the objective function. The constraints of Equation (6) ensure that the amount of flow assigned to each department duplicate does not exceed its capacity. The flow volume between departments are obtained from the product routings and product demands as follows:

$$v_{ijpt} = D_{pt}\sum_{k=1}^{S_p-1} \delta_{ipk}\delta_{jpk+1}, \quad (9)$$

where

$$\delta_{ipk} = \begin{cases} 1 & \text{if product } p \text{ is processed by department } i \text{ in stage } k, \\ 0 & \text{otherwise,} \end{cases}$$

and D_{pt} is the demand for product p in period t and S_p is the number of operations required by product p.

The model assumes that all department duplicates are of the same size. In practice this may not always hold, especially if we consider duplicates of the same department not containing the same number of machines. This problem can be addressed by dividing departments into small grids with equal area, say the size of a single machine, and assigning artificially large flows between grids of the same department duplicate so that they are always placed in adjoining locations. Alternative methods for incorporating departments of unequal size have been proposed (e.g., see Montreuil and Venkatadri (1991) and Yang and Peters (1998)). A discussion of the general merits and limitations of these approaches can be found in Kusiak and Heragu (1987), Bozer and Meller (1997) and Benjaafar and Sheikhzadeh (2000), among others.

Our criterion in designing layouts for each period is an aggregate measure of material handling and relayout cost over the entire planning horizon. However, this does not guarantee that the selected layouts would perform equally well in all periods. In fact, it is conceivable that selected layouts could perform poorly in certain periods. This may be unacceptable to plant managers who need to guarantee a certain level of performance consistency in each period. Such a guarantee could be achieved by including a robustness constraint that ensures that the material handling cost for each period is within a specified range of the optimal layout for that period. In other words, if we let z_t^* be the optimal material handling cost in period t, then the additional constraint is of the following form:

$$\sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} v_{nimjpt}\, x_{nikt}\, x_{mjlt}\, c_{kl} d_{kl} \le (1+\alpha) z_t^*, \tag{10}$$

where $0 \le \alpha \le 1$. The constraint guarantees the material handling cost of the selected layout for any period not to exceed the cost of the optimal layout for that scenario by a factor of $1+\alpha$. Obtaining z_t^* for each period t requires solving a special case of our original model with $r_{ikl} = 0$ for all i, k, and l.

Since our model assumes full flexibility in how flow volumes can be allocated among duplicates of the same department type, it can result in unbalanced workloads being placed on the different duplicates. Whereas this is always optimal from a material handling perspective it can lead to increased congestion and longer cycle times at the more utilized departments. The problem could be remedied by placing an upper limit on the level of allowed utilization of each department. This can be achieved by substituting the constraints of Equation (6) with constraints of the form:

$$\sum_{p=1}^{P}\sum_{i=0}^{N}\sum_{n=1}^{N_i} v_{nimjpt}\, t_{mjp} \le \gamma_{mj} C_{mj}, \tag{11}$$

where $0 \le \gamma_{mj} \le 1$.

In practice, it may also be desirable not to split orders that belong to the same product among duplicates of the same department. Order splitting can lead to smaller production batches and more frequent setups. It can also cause delays in shipping completed orders due to poor synchronization among individual batches of the same orders. Discouraging splitting of flows due to the same product among multiple department copies could be handled by adding another expression to the objective function that penalizes order splitting as follows:

$$\begin{aligned}\text{Min } z = &\sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} v_{nimjpt}\, x_{nikt}\, x_{mjlt}\, c_{kl} d_{kl}\\ &+\sum_{t=1}^{T-1}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{k=1}^{M}\sum_{l=1}^{M} x_{nikt}\, x_{nilt+1} r_{nikl}\\ &+\sum_{t=1}^{T}\sum_{p=1}^{P} b_{pt} \sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j} y_{nimjpt},\end{aligned} \tag{12}$$

and by introducing the following constraints:

$$v_{nimjpt} \le v_{ijpt}\, y_{nimjpt}, \tag{13}$$

where b_{pt} is a time-dependent order splitting penalty for product p and

$$y_{nimjpt} = \begin{cases} 1 & \text{if there exists a positive flow due to product } p \text{ between the } n^{\text{th}} \text{ departent duplicate of type } i \text{ and the } m^{\text{th}} \text{ department duplicate of type } j \text{ at period } t,\\ 0 & \text{otherwise.}\end{cases}$$

The choice of the penalty parameter b_{pt} is a managerial decision that depends on the perceived trade-off between the efficient flow allocation between department copies and the need to maintain order integrity.

In the case where there is ample processing capacity at each duplicate, the order splitting can be discouraged by restricting the assignment of each product to only one duplicate. This could be achieved for example by substituting constraints (4) with the following ones:

$$v_{ijpt}\, y_{nimjpt} = v_{nimjpt} \quad \forall n, i, m, j, p, t, \tag{14}$$

where

$$\sum_{n=1}^{N_i}\sum_{m=1}^{N_j} y_{nimjpt} = 1 \quad \forall i, j, p, t. \tag{15}$$

Note that not allowing order splitting would naturally give rise to an allocation not unlike the one obtained in a cellular layout, where each product is restricted at each stage of its processing to a specific machine or a group of machines. Furthermore, allowing order splitting at only a subset of the machines could be viewed as similar to partial machine sharing in cellular layouts. However, even for these cases, distributed and cellular layouts remain fundamentally different. In a distributed layout, there is no underlying cellular structure or a grouping of products. Unlike cellular layouts, products in a distributed layout may share some machines but could require different machines for most of their other processing.

3. Bounds on the optimal solution

There are two limiting cases to the dynamic distributed layout problem. The first is where rearrangement costs are insignificant, allowing us to solve a series of independent single-period layout problems. The second is where rearrangement costs are prohibitively high. In this case, no rearrangements take place, allowing us to combine all the flows from all the periods and solve a single-period layout problem. The first case provides us with the following lower bound:

$$LB_z^{(1)} = \sum_{t=1}^{T} z_t^{1^*}, \quad (16)$$

where

$$z_t^{1^*} = \sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} v^*_{nimjpt}x^*_{nikt}x^*_{mjlt}c_{kl}d_{kl}, \quad (17)$$

is the optimal material handling cost for period t and v^*_{nimjpt} and x^*_{nikt} denote respectively the optimal values for the flow allocations and layout assignments. This case also provides us with the following upper bound:

$$UB_z^{(1)} = \sum_{t=1}^{T} z_t^{1^*} + \sum_{t=1}^{T-1} z_t^{2^*}, \quad (18)$$

where

$$z_t^{2^*} = \sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{k=1}^{M}\sum_{l=1}^{M} x^*_{nikt}x^*_{nil(t+1)}r_{nikl}, \quad (19)$$

is the rearrangement cost between periods t and $t+1$ given the department locations implied by Equation (18). The second case gives us a second upper bound:

$$UB_z^{(2)} = \sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} \overline{v^*_{nimjp}}x^*_{nik}x^*_{mjl}c_{kl}d_{kl}, \quad (20)$$

where

$$\sum_{n=1}^{N_i}\sum_{m=1}^{N_j} \overline{v^*_{nimjp}} = \sum_{t=1}^{T} v_{ijpt} \quad \forall i, j, \text{and } p, \quad (21)$$

and the x^*_{nik}'s and $\overline{v^*_{nimjp}}$'s are the optimal solution for a single-period problem with flows $v_{ijp} = \sum_{t=1}^{T} v_{ijpt}$.

The relative difference $\rho = (UB_z^{(2)} - LB_z^{(1)})/LB_z^{(1)}$, which we term the *robustness gap*, provides a measure of the maximum relative amount of benefit that is forgone if a fixed layout is adopted for all periods. In settings where ρ is small, a robust layout is close to the optimal one for each period. If, on the other hand, ρ is large, then some rearranging of the layout between periods could be desirable. In Section 5, we show that distributed layouts tend to exhibit relatively low values of ρ. The parameter ρ is affected by the degree to which material flow patterns change from period to period. In environments where material flow patterns do not change significantly, ρ is small, while in highly variable environments, ρ is large.

Additional useful lower bounds can be obtained from bounds on the optimal solution for the single-period problem. A lower bound for the optimal solution in each period can be obtained by solving the following assignment problem (note that for the sake of simplicity, we drop the index t):

$$\text{(PA)} \quad \text{Min } z = \sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} v_{nimjp}w_{(nimj)(kl)}c_{kl}d_{kl}, \quad (22)$$

subject to

$$\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j} w_{(nimj)(kl)} = 1 \quad \forall k, l, \quad (23)$$

$$\sum_{k=1}^{M}\sum_{l=1}^{M} w_{(nimj)(kl)} = 1 \quad \forall i, n, j, m, \quad (24)$$

$$w_{nimjkl} \in 0, 1 \quad \forall i, n, j, m, k, l, \quad (25)$$

and the constraints of Equations (4), (5), (6), and (8).

In this formulation, the product $x_{nik}x_{mjl}$ is substituted by w_{nimjkl}. Problem (PA) is a relaxation of the single-period version of problem (P). Pairs of department duplicates are assigned to pairs of locations. Consequently, more than one department can be assigned to one location and the solution of problem (PA) can be infeasible in problem (P). Problem (PA) is still combinatorially hard. However, there are two important instances that are easily solvable. The first is when capacity constraints are relaxed. In this case, we can show that the problem reduces to the following simple linear assignment problem:

$$\text{(PB)} \quad \text{Min } z = \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{M}\sum_{l=1}^{M} v_{ij}w_{(ij)(kl)}c_{kl}d_{kl}, \quad (26)$$

subject to

$$\sum_{i=1}^{N}\sum_{j=1}^{N} w_{(ij)(kl)} = 1 \quad \forall k, l, \quad (27)$$

$$\sum_{k=1}^{M}\sum_{l=1}^{M} w_{(ij)(kl)} = 1 \quad \forall i, j, \quad (28)$$

$$w_{ijkl} \in 0, 1 \quad \forall i, j, k, l, \quad (29)$$

where $v_{ij} = \sum_{p=1}^{P} v_{ijp}$. An optimal solution to the above linear assignment problem can be obtained via a greedy algorithm in which we assign the largest v_{ij} to the pair of locations (k, l) with the smallest cost-distance product

$c_{kl}d_{kl}$, the second largest v_{ij} to the pair with second smallest $c_{kl}d_{kl}$, and so forth. Reintroducing the period index t, a second lower bound on the total optimal cost can be obtained as $LB_z^{(2)} = \sum_{t=1}^{T} LB_{zt}^{(2)}$, where $LB_{zt}^{(2)}$ refers to the above single-period lower bound. The second instance for which problem (PA) is easily solvable is when the copies of each department type are identical (i.e., $C_{mj} = C_j$ and $t_{mjp} = t_{jp}$ for all j and p). The optimal solution can then be obtained using the approach suggested by Urban *et al.* (2000). Although the lower bounds discussed in this section are generally loose, we show later that they can be remarkably tight for distributed layouts.

4. Solution procedure

An instance of the dynamic distributed layout problem in Equations (1)–(6) is the Quadratic Assignment Problem (QAP), a well-known NP-hard problem (Pardalos and Wolkowicz, 1994) (the QAP corresponds to the special case of $N_i = 1$ for all i, $r_{ikl} = 0$ for all i, k and l, and $t = 1$). Therefore, the dynamic distributed layout problem is also NP-hard, which means that finding optimal solutions for large problems is generally difficult (the largest solved QAP problem is of the order of 30 departments). In our case, since both department locations and flow volume allocations are decision variables, the size of solvable problems would be even smaller. In Appendix 1, we provide an exact solution procedure for the distributed layout based on a branch-and-bound algorithm. However, the usefulness of this procedure is limited to small problems.

In the remainder of this section, we offer a decomposition-based heuristic that we show to perform well relative to lower bounds. In practice, solution optimality in the design of dynamic plant layouts is not critical since there is often uncertainty surrounding the value of design parameters such as future demands, department capacities, and rearrangement costs. Fortunately, distributed layouts are quite robust to these uncertainties. As we show in Section 5, due to this built-in robustness, the difference between optimal and suboptimal layouts is relatively small.

Our heuristic approach is based on an iterative procedure in which we decompose the problem into two subproblems: (i) a facility layout subproblem; and (ii) a flow allocation subproblem. A solution is obtained by iteratively solving for a facility layout problem with fixed flows followed by a flow allocation problem with a fixed layout. The procedure can be summarized as follows:

Step 1. Given a layout for each period, we find a minimum cost flow allocation between department duplicates for each of these periods.

Step 2. Given a flow allocation between department duplicates for each period, we find optimal layouts for these periods.

We alternate between Steps 1 and 2, until no further improvements to the solution are possible. Whereas it does not guarantee optimality, this approach does satisfy the following necessary conditions for optimality. For the obtained layout, the solution cannot be improved by a change in flow allocation. Similarly, for the obtained flow allocation, the solution cannot be improved by a change in the layout.

The flow allocation subproblem that must be solved in Step 1 is given by:

$$\text{Min } z = \sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} v_{nimjpt}\, x_{nikt}\, x_{mjlt}\, c_{kl} d_{kl}, \tag{30}$$

subject to the constraints of Equations (4), (5), (6), and (8). The values of x_{nikt} and x_{mjlt} are known and only the flow volume allocations v_{nimjpt} are decision variables. Since the objective function and the constraints are all linear in the decision variables, we have a linear program which can be solved in polynomial time. The problem can be further simplified by noting that it can be decomposed into T separable problems, one for each period, which can be solved independently without affecting the overall optimality. The problem can also be viewed as a multi-commodity network flow problem with side constraints and solved using specialized algorithms for this class of problems (see McBride (1998) for a recent survey).

The layout subproblem that needs to be solved in Step 2 is given by:

$$\text{Min } z = \sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} v_{nimjpt}\, x_{nikt}\, x_{mjlt}\, c_{kl} d_{kl} + \sum_{t=1}^{T-1}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{k=1}^{M}\sum_{l=1}^{M} x_{nikt}\, x_{nilt+1}\, r_{nikl}, \tag{31}$$

subject to the constraints of Equations (2), (3), and (7). Here the flow allocations are known and only the department locations are decision variables. Unlike the flow allocation subproblem, we cannot separate the layout design subproblem into independent problems after each period since rearrangement costs are dependent on the layouts of the preceding and current period. The layout design problem is the Dynamic Layout Problem (DLP) which is NP-hard (Pardalos and Wolkowicz, 1994). The problem can be solved exactly using a branch-and-bound algorithm for small-sized problems. For larger problems, a heuristic approach is needed. Several heuristics have been proposed for the DLP, including 2-opt (Urban, 1993), genetic algorithms (Kochhar and Heragu, 1999) and tabu-search (Kaku and Mazzola, 1997). In this study, we use a modified version of a 2-opt heuristic, which we found to perform reasonably well against known lower and upper bounds.

In implementing the decomposition approach, we experimented with two search algorithms A1 and A2. The two algorithms carry out the decomposition process in slightly different ways leading to some differences in the computational effort and solution quality.

Algorithm A1

Step 1. Set $i = 1$ and generate an initial layout $\mathbf{x}^*(t, i)$, a department location matrix, for each period t.
Step 2. Determine an optimal flow allocation matrix $\mathbf{v}^*(t, i)$ for each period t given layout $\mathbf{x}^*(t, i)$ (this requires solving a linear program).
Step 3. Set $i = i + 1$ and find a global 2-opt layout $\mathbf{x}^*(t, i)$ for each period t given flow matrix $\mathbf{v}^*(t, i - 1)$.
Step 4. Determine the optimal flow allocation $\mathbf{v}^*(t, i)$ for layout $\mathbf{x}^*(t, i)$.
Step 5. Let $z^*(i)$ correspond to total cost given matrices $\mathbf{v}^*(t, i)$ and $\mathbf{x}^*(t, i)$. If $z^*(i) < z^*(i - 1)$ go back to Step 3. Otherwise, Stop.

Step 3 requires finding a global 2-opt layout, which means that there is not a pair-wise interchange of the locations of any two departments in any period that could improve the solution. Step 3 requires carrying out a series of pair-wise interchanges. There are a variety of ways in which we can sequence these interchanges. To minimize period-bias, we choose to carry these out in a series of iterations, where in the first iteration we evaluate only $M(M - 1)$ interchanges from the layout in each period. We then implement the interchange, out of the $T(M(M - 1))$ interchanges, that reduces our total cost the most. This is followed by a new iteration which repeats the same process for a maximum of $3(M(M - 1))$ interchanges. These interchanges involve the period for which the layout has been modified in the previous iteration, and the period that immediately precedes and succeeds it. The process continues until no further improvements are possible.

A potential limitation of algorithm A1 is that interchanges carried out in Step 3 are evaluated using a fixed flow allocation. The true cost of the interchange may not be entirely reflected since we use a suboptimal flow allocation to evaluate its cost. An alternative would be to evaluate the optimal flow allocation associated with each interchange and use this flow allocation in calculating its cost. We implemented this approach in an algorithm we call A2. In algorithm A2, we retain only Step 3 of algorithm A1. However, for every pair-wise interchange carried out in this step, we compute the corresponding optimal flow allocation, which requires solving a linear program. The 2-opt solution we retain at the end of this process is always flow-optimal. Experimentation with this approach shows that whereas algorithm A2 can be more computing intensive, it generally leads to higher quality solutions than algorithm A1.

For both algorithms A1 and A2, the quality of the solution is sensitive to our choice of initial layout. In order to limit initial solution bias, we generate solutions with multiple initial layouts. Among the initial layouts, we always include the (heuristic) solutions corresponding to the lower and upper bounds $LB_z^{(1)}$ and $UB_z^{(2)}$. We find that using solutions for these bounds as initial layouts tends to yield better solutions than those obtained using randomly generated layouts.

5. Numerical results and empirical investigations

In order to: (i) examine the quality of our solution; (ii) assess the computational effectiveness of our solution procedure; and (iii) study the effect of different system parameters, especially department duplication and distribution, we conducted a series of computational experiments using randomly generated examples with varying layout sizes, levels of department duplication, number of products, product routings, demand distributions, and period lengths.

Our solution algorithm, along with a data generating procedure, was implemented in a program application written in C and interfaced with the commercial optimization solver Cplex version 7.0. The solver is called from the main application to solve the optimal flow allocation subproblem. The implementation platform is a Sun Ultra Sparc workstation running the Solaris operating system. We have made no attempt at this stage to optimize the computational performance of the algorithm. Since layout design decisions are long-term in nature, computational performance is usually not critical and even several hours of CPU can be acceptable.

In our experiments we consider layout sizes that range between 16 and 48 departments and between four and eight department types. Several levels of unit rearrangement costs, demand distributions, and product routings were also considered. In order to examine the impact of department disaggregation and distribution, we carried out a series of experiments of layouts with varying levels of department disaggregation. For instance in a layout consisting of 48 grids and six department types, we consider four levels of department disaggregation. In level 1, there is a single copy of each department type (the eight processors of each department type are placed in adjoining locations). In level 2 (3), each department type is assigned two (four) copies, each with four (two) processors placed in adjoining locations. In level 4, there are eight copies of each department, which means that individual processors of the same type can be placed independently of each other.

For functional (partially distributed) layouts, we restrict all (some) of the departments of the same type to be in adjacent locations and require that these aggregated departments have reasonably compact shapes. We do so by segmenting the floor space into bands and the placement of departments is restricted within these bands. An initial layout is formed by starting at the upper or lower left-hand corner of the first band and "sweeping" the bands in a serpentine fashion by placing departments according to a prespecified fill sequence, also called a Space-Filling Curve (SFC)

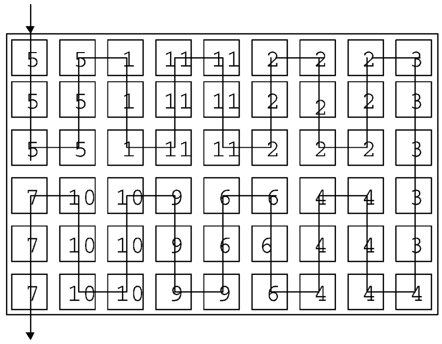

Fig. 3. An example SFC for a functional layout.

(Bozer *et al.*, 1994). An example SFC is shown in Fig. 3. The SFC and the number of bands are user-specified. For our examples, we restrict the number of bands to two (experiments with three and four bands lead generally to a poorer performance). This approach allows us to exchange the location of functional or partially distributed departments regardless of whether or not they are of the same size.

An alternative approach is to assign large artificial flows between copies of the same department type. Our preliminary experimentation shows that the use of SFCs for these layouts provides higher quality solutions in less time. In fact, the use of artificial flows requires evaluating exchanges of individual processors (including those belonging to the same department) rather than entire subdepartments, as is possible with the SFC-based approach. More importantly, the use of SFCs guarantees compactness of the resulting departments.

For a fixed flow allocation, functional and partially distributed layouts are obtained using the same exchange heuristic used for distributed layouts. Similarly, flow allocation is carried out with respect to individual department copies within the aggregate department using the flow allocation model. Rectilinear distances between the centroids of individual department copies are used.

We generated several hundred data sets for a variety of layout sizes, demand distributions, product numbers, and routings. For each data set, we generated a solution using our decomposition procedure. We also obtained an exact value for the lower bound $LB_z^{(2)}$ and approximations for $LB_z^{(1)}$, $UB_z^{(1)}$, and $UB_z^{(2)}$ which we denote respectively as $\widehat{LB}_z^{(1)}$, $\widehat{UB}_z^{(1)}$, and $\widehat{UB}_z^{(2)}$. They are approximations since the associated layout design problems cannot be solved optimally for the size of problems we consider. Note that $\widehat{UB}_z^{(1)}$ and $\widehat{UB}_z^{(2)}$ are valid upper bounds since they are upper bounds on $UB_z^{(1)}$ and $UB_z^{(2)}$. Although $\widehat{LB}_z^{(1)}$ is not a guaranteed lower bound, we found it for most distributed layouts to be remarkably close to $\widehat{LB}_z^{(2)}$, which is a relatively loose bound. Representative results are shown in Tables 1 and 2.

Table 1. Minimum total cost, lower bound and upper bounds for a sample data set

Sample	*Distributed*	*Partial*	*Functional*	*Random*	$\widehat{LB}_z^{(1)}$	$\widehat{UB}_z^{(1)}$	$\widehat{UB}_z^{(2)}$	$LB_z^{(2)}$
1	14 618	17 754	26 233	19 060	14 111	15 805	15 111	13 812
2	5176	5989	8497	6440	4893	5534	5463	4782
3	5188	5799	9071	6562	4958	5528	5368	4902
4	7227	8004	11 334	9070	7056	7711	7926	7056
5	8528	10 167	15 669	10 656	8363	9552	8558	8268
6	2874	3129	4499	3320	2844	3003	2926	2844
7	1264	1470	2104	1765	1257	1484	1319	1235
8	7877	8600	12 548	9492	7794	8393	8086	7794
9	5328	5930	8295	7168	5223	5983	5413	5223
10	3816	4760	6798	4917	3536	4210	3926	3436
11	4505	5087	6798	6003	4265	4931	4665	4227
12	2322	2545	3788	3261	2252	2534	2672	2252
13	3577	4185	5885	4702	3474	3703	3867	3474
14	3251	3552	5054	4465	3050	3521	3430	2850
15	3249	3922	5623	4056	2919	3352	3969	2820
16	755	857	1262	906	654	755	1101	654
17	2873	3440	4721	3763	2506	3157	3456	2506
18	4372	5542	7602	6120	4330	4562	4396	4200
19	4529	5503	8147	5909	4409	4937	4534	4200
20	4621	5900	9736	5656	4506	4738	4626	4200
21	4578	5710	8732	5954	4368	4712	4588	4200
22	4690	6518	9914	5427	4508	4690	5310	4200
23	3040	3680	5560	4488	3000	3488	3130	3000
24	3180	4262	6472	4606	3000	3408	3380	3000
25	442	535	769	527	418	472	443	406
26	354	439	736	400	324	377	369	324

Table 2. Robustness gap, percentage differences between different layout configuration results, and lower and upper bounds for example data sets

Sample	% gap $\frac{(\widehat{UB}_z^{(1)} - \widehat{LB}_z^{(1)})}{\widehat{LB}_z^{(1)}}$	% difference $\frac{(Func-Dist)}{Dist}$	% difference $\frac{(\widehat{LB}_z^{(1)} - LB_z^{(2)})}{LB_z^{(2)}}$	% difference $\frac{(Dist-\widehat{LB}_z^{(1)})}{\widehat{LB}_z^{(1)}}$	% difference $\frac{(Dist-LB_z^{(2)})}{LB_z^{(2)}}$	% difference $\frac{(Rand-Dist)}{Dist}$
1	10.7	44.3	2.2	5.5	3.5	23.3
2	11.6	39.1	2.3	7.6	5.5	19.6
3	10.3	42.8	1.1	5.5	4.4	20.9
4	8.5	36.2	0.0	2.4	2.4	20.3
5	12.4	45.6	1.1	3.0	1.9	20.0
6	5.3	36.1	0.0	1.0	1.0	13.4
7	15.3	39.9	1.8	2.3	0.6	28.4
8	7.1	37.2	0.0	1.1	1.1	17.0
9	12.7	35.8	0.0	2.0	2.0	25.7
10	16.0	43.9	2.9	10.0	7.3	22.4
11	13.5	33.7	0.9	6.2	5.3	25.0
12	11.1	38.7	0.0	3.0	3.0	28.8
13	6.2	39.2	0.0	2.9	2.9	23.9
14	13.4	35.7	7.0	12.3	6.2	27.2
15	12.9	42.2	3.5	13.2	10.2	19.9
16	13.4	40.2	0.0	13.4	13.4	16.7
17	20.6	39.1	0.0	12.8	12.8	23.7
18	5.1	42.5	3.1	3.9	1.0	28.6
19	10.7	44.4	5.0	7.3	2.6	23.4
20	4.9	52.5	7.3	9.1	2.5	18.3
21	7.3	47.6	4.0	8.3	4.6	23.1
22	3.9	52.7	7.3	10.4	3.9	13.6
23	14.0	45.3	0.0	1.3	1.3	32.3
24	12.0	50.9	0.0	5.7	5.7	31.0
25	11.4	42.5	3.0	8.1	5.4	16.1
26	14.1	51.9	0.0	8.5	8.5	11.5
Average	10.9	42.3	2.0	4.6	6.4	22.1

As we can see, the difference in cost between our solution and $\widehat{LB}_z^{(1)}$ is on average 5% and the difference between our solution and $LB_z^{(2)}$ is on average 7%. Given that these lower bounds are generally not tight, the results are encouraging. In order to further benchmark our layout design procedure, we generated layouts using a purely random assignment of departments to locations in each period. One hundred such layout combinations are randomly generated. For each layout combination, we obtain the optimal flow allocation in each period. The layout solution with the lowest expected cost is then selected (see Tables 1 and 2).

5.1. *Preliminary results*

As we can see from Table 2, distributed layouts vastly outperform functional layouts with costs being an average of 42% lower for the distributed layout. This is of course not entirely surprising since the functional layout is a constrained version of the distributed one. However, it is somewhat surprising to observe how close the performance of the distributed layouts is to that of the lower bounds, especially $LB^{(2)}$ (computed exactly) and which is generally a loose bound. Distributed layouts also outperform the best found random layout. This difference is smaller than expected, with the cost of distributed layouts being on average 22% lower than the best random one. Random layouts tend to exhibit a high degree of department distribution and therefore realize most of the associated benefits. This seems to support the intuition that, as long as departments are sufficiently dispersed, the benefit of optimally assigning departments to locations is relatively small (this also supports the use of heuristics in designing distributed layouts). This is further confirmed by values of the robustness gap which range from 4 to 20% with an average of 11% (Table 2). Comparisons with partially distributed layouts reveal that the difference between partially and fully distributed layouts is relatively small. In the next section we further explore the effect of varying degrees of department duplication and distribution.

5.2. *The effect of department duplication*

In order to examine the effect of department duplication on layout performance, we carried out a series of experiments with layouts of varying levels of department disaggregation and distribution. Representative results from a

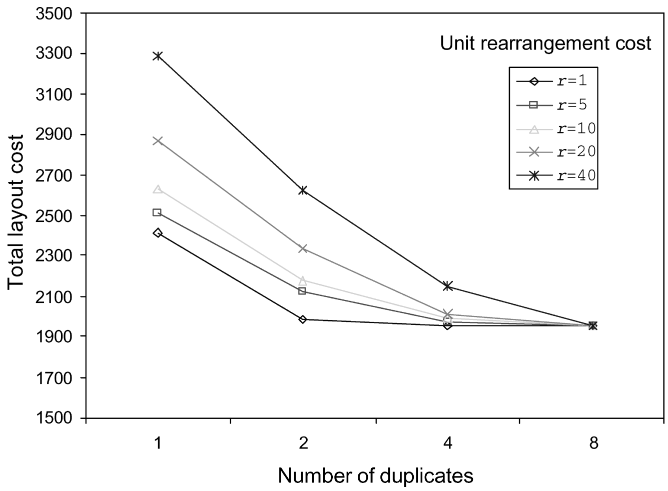

Fig. 4. The effect of department duplication on the total layout cost for different unit rearrangement costs. Each data point is an average of 50 experiments for a system with three products, randomly generated routings with random sequence lengths from three to eight and randomly generated demand from a uniform distribution with range (50, 250).

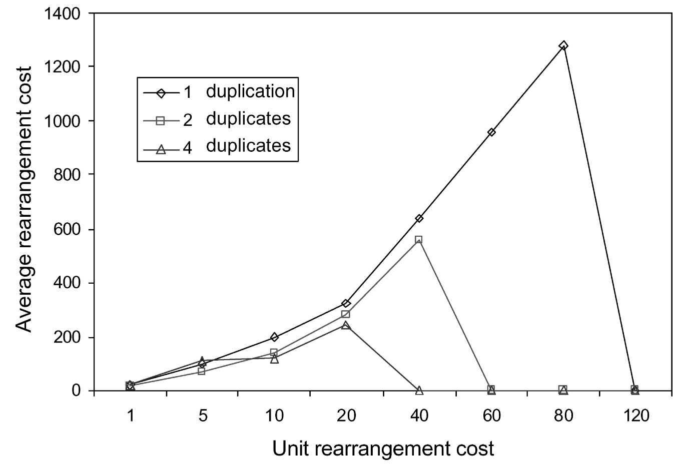

Fig. 5. The effect of unit rearrangement cost on the total layout cost for different duplicate levels.

layout with 48 departments equally divided among six department types (eight department copies per department type) are shown in Fig. 4. Department duplication is always beneficial, with up to a 40% cost reduction. However, the effect of duplication is of the diminishing kind with most of the benefits realized with the initial disaggregation of departments into two subdepartments (for example, an increase from one to two results in an average reduction of 22% in total cost whereas a further increase from two to four yields a less than 11% additional decrease in cost). This confirms observations made by Benjaafar and Sheikhzadeh (2000) for the single-period problem. This also seems to suggest that in practice, full department disaggregation and distribution would rarely be justified.

5.3. *The effect of rearrangement costs*

As seen from Fig. 4, the benefits of distributed layouts are generally increasing in unit rearrangement costs. This effect is best seen by considering only the rearrangement cost component in the total optimal cost. As shown in Fig. 5, distributed layouts incur lower rearrangement costs (especially when unit rearrangements costs are high) and have a lower Rearrangement Threshold (RT), where RT refers to the unit rearrangement cost above which a fixed (robust) layout is selected for all periods so that rearrangement costs become zero. These effects can be explained as follows. A distributed layout can respond to changes in flow requirements by moving individual department copies. In contrast, a functional layout must relocate all copies within a department to enable any rearrangement. A distributed layout tends to be robust, making the benefits from layout rearrangements relatively small. Therefore, relayout becomes uneconomical for lower values of rearrangement cost than for a functional layout.

5.4. *The effect of flow variability*

Intuition suggests that distributed layouts are more desirable when variability in the flows is high. We examine the effect of three potential sources of variability on the value of distributed layouts: (i) demand variability; (ii) routing variability; and (iii) demand volatility. We measure demand variability by the extent to which demand for each product changes from one period to the next, or equivalently by the change in the relative composition of the product mix. If all products change by similar proportions (positive or negative), then there would be little impact on layout since flow patterns would remain unchanged. However, if the relative demand composition changes so that new products become more or less dominant than in previous periods, flow patterns could significantly change, and the effect on layout could be significant. A simple measure of this type of variability (i.e., product mix dissimilarity) between two periods t and t' is given by the following:

$$\delta_{tt'} = \sum_{p=1}^{P} |\alpha_{pt} - \alpha_{pt'}|/P, \qquad (32)$$

where $\alpha_{pt} = D_{pt}/\sum_{i=1}^{P} D_{it}$ is the percentage of total demand in period t due to product p. Note that the ratio ranges from zero to one, with the ratio being zero when the relative product mix remains unchanged.

Although the above measure accounts for an important source of flow variability, it does not capture the difference in the routing sequences between the product produced in different periods. This variability could have a significant impact on layout since it ultimately determines the degree to which flow patterns change from period to period. A possible measure of routing dissimilarity between period t and t', to which we call (loosely) routing variability is given by the ratio:

$$\beta_{tt'} = 1 - \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} I_{ijt} I_{ijt'}}{\sum_{i=1}^{N}\sum_{j=1}^{N} I_{ijt} + \sum_{i=1}^{N}\sum_{j=1}^{N} I_{ijt'} - \sum_{i=1}^{N}\sum_{j=1}^{N} I_{ijt} I_{ijt'}}, \quad (33)$$

where

$$I_{ijt} = \begin{cases} 1 & \text{if there is a positive flow between departments of types } i \text{ and } j \text{ in period } t, \\ 0 & \text{otherwise.} \end{cases}$$

Note that the ratio ranges from zero to one, with zero corresponding to the case where the two periods do not share any flow similarities.

The third type of variability, which we call volatility, is related to the frequency with which demand changes. That is, volatility is determined by the length of each period in a fixed planning horizon or, equivalently, by the number of periods within the horizon. For example, an environment where demand changes monthly is more volatile than one where demand changes quarterly.

The effects of these three types of variability for different levels of department duplication are shown in Figs. 6, 7, and 8. The design of experiments associated with these figures is described in Appendix 2. As we can see, higher variability in all three cases increases the robustness gap (the maximum relative improvement a robust layout would forgo by adopting a fixed layout for all periods) regardless of duplication level. This is of course not surprising since higher variability makes relayout more desirable. However, it is interesting to see that the differences in the robustness gaps between distributed and functional layouts, and

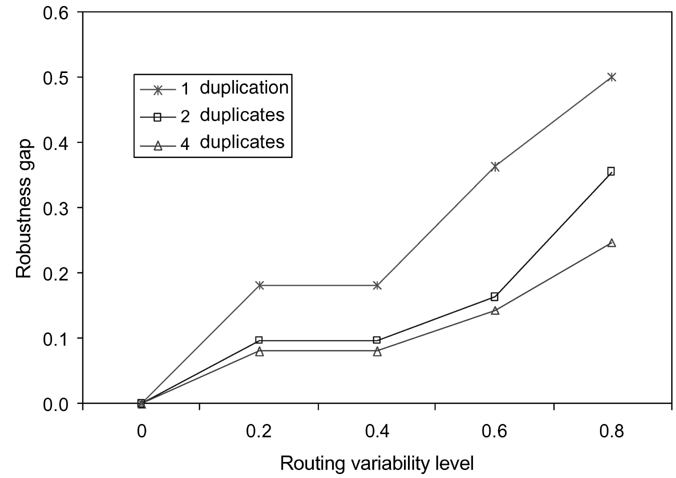

Fig. 7. The effect of routing variability on the robustness gap for different duplicate levels.

distributed and partially distributed ones, are higher when variability is high. This supports the intuition that distributed layouts are more valuable when flow variability is high.

5.5. *The effect of product variety*

It is tempting to assume that higher product variety (i.e., number of products produced in each period) induces higher flow variability and therefore makes distributed layouts more desirable. Closer examination reveals that this is not always true and that the reverse effect (i.e., distributed layouts becoming relatively less desirable with higher product variety) is more likely. In particular, observe that with increased product variety, given a fixed level of overall demand, there is an increased likelihood that most paths

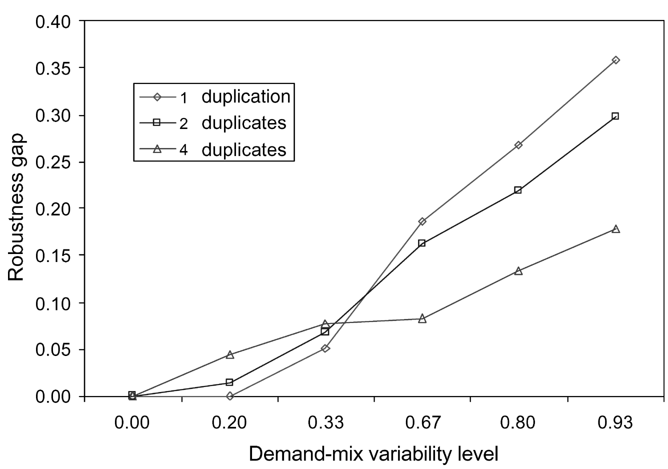

Fig. 6. The effect of demand mix volatility on the robustness gap for different duplicate levels.

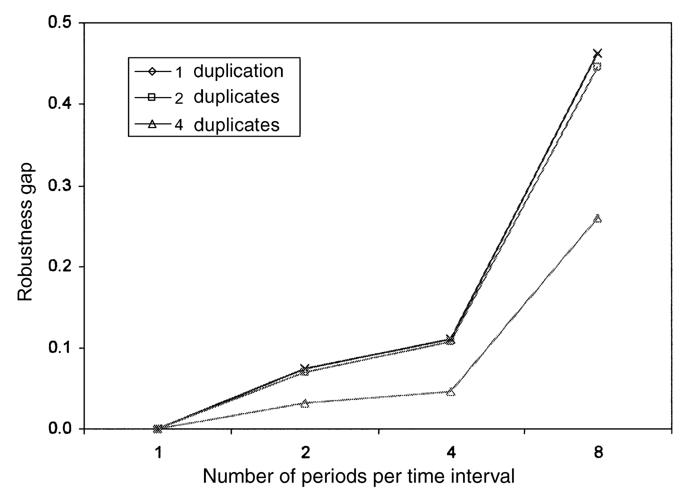

Fig. 8. The effect of period length on the robustness gap for different duplicate levels.

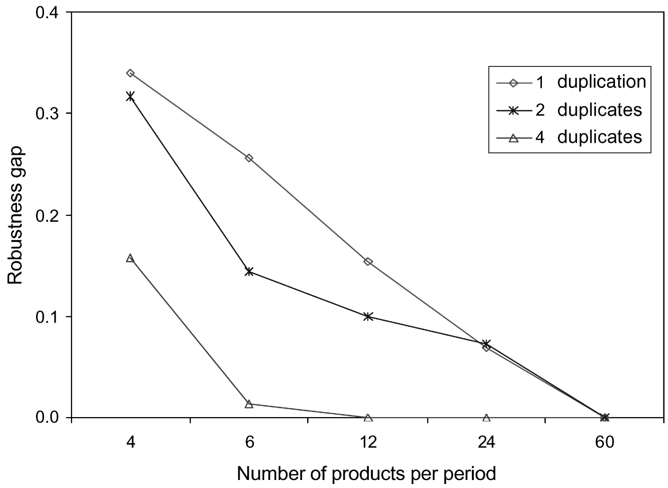

Fig. 9. The effect of demand volatility on the robustness gap for different duplicate levels.

between pairs of different department types would be used. In fact, in the limit case, all paths become equally likely to be used. Hence, higher product variety diminishes flow variability from period to period. In turn, this reduces the need to rearrange layouts between periods even if the adopted layout is a functional or a partially distributed one. These effects are confirmed by our numerical results. In the examples shown in Fig. 9, higher product variety leads to a smaller robustness gap and to a smaller difference in this gap between layouts with different duplication levels. Each data point is an average of 30 experiments, where in each experiment we fix the overall demand D_t in each period but vary the number of products so that $D_{pt} = D_t/P$. The systems are otherwise similar to those described in Appendix 2.

6. Conclusions and extensions

In this paper, we have shown how disaggregating functional departments into smaller subdepartments that are then distributed throughout the plant can significantly improve performance in a multi-period setting by: (i) improving flow efficiency within each period; and (ii) reducing the need for layout rearrangement between periods. We found that distributed layouts are particularly valuable when variability is high or product variety is low. More importantly, we found that most of the benefits of a distributed layout are realized with relatively few duplicates of each department type, which means there would rarely be a need to fully disaggregate functional departments. Because of the robustness that distributed layouts typically exhibit (a distributed layout is less vulnerable to changes in production requirements), optimizing the layout in each period carries significantly less value than it does for functional layouts. Consequently, a heuristic layout approach (coupled with an optimal flow allocation) tends to be sufficient.

There are several possible avenues for future research. In the current model, we do not account for the fact that there might be a cost associated with disaggregating and distributing functional departments. For example, there might be loss of economies of scale due to duplication of necessary support infrastructure that is typically shared by a consolidated functional department, such as operators, computer control systems, loading/unloading areas, and waste disposal facilities. An extended facility design model would allow for the number and size of department duplicates to be decision variables. For example, the objective function could be reformulated to capture benefits from department consolidation as follows:

$$\text{Min } z = \sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} v_{nimjpt}\, x_{nikt}\, x_{mjlt}\, c_{kl} d_{kl} + \sum_{t=1}^{T-1}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{k=1}^{M}\sum_{l=1}^{M} x_{nikt}\, x_{nilt+1} r_{nikl} - \sum_{t=1}^{T-1}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{k=1}^{M}\sum_{l=1}^{M} x_{nikt}\, x_{milt} I_{kl} e_{nmi}, \quad (34)$$

where the new term accounts for benefits (cost savings) due to having duplicates of same type adjacent to each other with

$$I_{kl} = \begin{cases} 1 & \text{if location } k \text{ and } l \text{ are adjacent,} \\ 0 & \text{otherwise,} \end{cases}$$

and e_{nmi} is the benefit of having both duplicates n and m of type i in adjacent locations. In addition to capturing the value of adjacently locating departments of the same type, this formulation has the advantage of not introducing any additional decision variables. The parameters I_{kl} can be obtained from the location grid matrix while e_{nmi} are estimates of cost reduction due to having two duplicates of the same type in adjoining locations. The reformulation also has the advantage of favoring compact departments by assigning higher benefits if a duplicate is adjacent to multiple departments of the same type.

In the current model, we allow all department duplicates to be assigned workload even when the demand could be handled by only a subset of these duplicates. This ignores activation costs that arise in some settings whenever a department duplicate is used. An extended model could include a fixed cost associated with using each department copy of each type in each period. This could then lead to layout solutions where some department duplicates are idled in periods where demand is sufficiently low. A possible reformulation would include the introduction of new decision variables:

$$y'_{nit} = \begin{cases} 1 & \text{if there exists a positive flow into/from duplicate } n \text{ of type } i \text{ in period } t, \\ 0 & \text{otherwise,} \end{cases}$$

and the objective function:

$$\text{Min } z = \sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{j=1}^{N}\sum_{m=1}^{N_j}\sum_{k=1}^{M}\sum_{l=1}^{M} v_{nimjpt}\, x_{nikt}\, x_{mjlt}\, c_{kl} d_{kl} + \sum_{t=1}^{T-1}\sum_{i=1}^{N}\sum_{n=1}^{N_i}\sum_{k=1}^{M}\sum_{l=1}^{M} x_{nikt}\, x_{nilt+1} r_{nikl} + \sum_{t=1}^{T}\sum_{i=1}^{N}\sum_{n=1}^{N_i} K_{nit}\, y'_{nit}, \quad (35)$$

where K_{nit} is a fixed cost for using duplicate n of type i in period t. In order to ensure that a duplicate is assigned workload only when it is activated, we must also include the following constraint:

$$\sum_{p=1}^{P}\sum_{j=1}^{N}\sum_{m=1}^{N_j} v_{nimjpt} \leq M y'_{nit}, \quad (36)$$

where M is a large number bounded by $\sum_{p=1}^{P}\sum_{k=1}^{S_p} D_{pt}\delta_{ipk}$.

In our numerical experiments, we have investigated the effect of department duplication assuming the same number of duplicates for each department type. In some settings, different departments may have different numbers of machines. Upon disaggregation, this could lead to different numbers of duplicates for different departments. In other settings, certain departments are processing bottlenecks. In that case, having a single or a few shared duplicates for the bottleneck department is desirable in achieving better load balancing among individual machines. For these cases and others, it is of interest to investigate the extent to which differences in the level of duplication among departments would affect layout performance.

References

Askin, R.G., Lundgren, N.H. and Ciarallo, F. (1997) A material flow based evaluation of layout alternatives for agile manufacturing, in *Progress in Material Handling Research*, Graves, R.J., McGinnis, L.F., Medeiros, D.J., Ward, R.E. and Wilhelm, M.R. (eds.), Material Handling Institute, Ann Arbor, MI, pp. 71–90.

Balakrishnan, J. (1993) The dynamics of plant layout. *Management Science*, **39**(5), 654–655.

Balakrishnan, J. and Cheng, C.H. (1998) Dynamic layout algorithms: a state of the art survey. *Omega*, **26**(4), 507–521.

Batta, R. (1987) Comments on the dynamics of plant layout. *Management Science*, **33**(8), 1065.

Benjaafar, S. (1995) Machine sharing in cellular manufacturing systems, in *Planning, Design, and Analysis of Cellular Manufacturing Systems*, Parasei, H.R., Kamrani, A.K. and Liles, D.H. (eds.). Elsevier Science B.V., 203–228.

Benjaafar, S. (2002) Modeling and analysis of congestion in the design of facility layouts. *Management Science*, **48**(5), 679–704.

Benjaafar, S., Heragu, S. and Irani, S.A. (2002) Next generation factory layouts: research challenges and recent progress. *Interfaces*, **32**(6), 58–76.

Benjaafar, S. and Sheikhzadeh, M. (2000) Design of flexible plant layouts. *IIE Transactions*, **32**(5), 309–322.

Bozer, Y. and Meller, R.D. (1997) A reexamination of the distance-based facility layout problem. *IIE Transactions*, **29**(7), 549–560.

Bozer, Y., Meller, R.D. and Erlebacher, S.J. (1994) An improvement-type layout algorithm for single and multiple floor facilities. *Management Science*, **40**(7), 918–932.

Drolet, J.R. (1989) Scheduling virtual cellular manufacturing systems. PhD thesis, School of Industrial Engineering, Purdue University, West Lafayette, IN.

Heragu, S.S. (1994) Group technology and cellular manufacturing. *IEEE Transactions on Systems, Man and Cybernetics*, **24**(2), 203–215.

Irani, S.A., Cavalier, T.M. and Cohen, P.H. (1993) Virtual manufacturing cells: exploiting layout design and intercell flows for the machine sharing problem. *International Journal of Production Research*, **31**(4), 791–810.

Irani, S.A. and Huang, H. (2000) Custom design of facility layouts for multi-product facilities using layout modules. *IEEE Transactions on Robotics and Automation*, **16**(3), 259–267.

Kaku, B.K. and Mazzola, J.B. (1997) A tabu-search heuristic for the dynamic plant layout problem. *INFORMS Journal on Computing*, **9**, 374–384.

Kochhar, J.S. and Heragu, S.S. (1999) Facility layout design in a changing environment. *International Journal of Production Research*, **37**(11), 2429–2446.

Kouvelis, P. and Kiran, A.S. (1991) Single and multiple period layout models for automated manufacturing systems. *European Journal of Operational Research*, **52**(3), 300–314.

Kouvelis, P., Kurawarwala, A.A. and Gutierrez, G.J. (1992) Algorithms for robust single period and multiple period layout planning for manufacturing systems. *European Journal of Operational Research*, **63**(2), 287–303.

Kusiak, A. and Heragu, S.S. (1987) The facility layout problem. *European Journal of Operational Research*, **27**(3), 229–251.

McBride, R. (1998) Advances in solving the multicommodity-flow problem. *Interfaces*, **28**(2), 32–41.

Meller, R. and Gau, K.Y. (1996) The facility layout problem: recent and emerging trends and perspectives. *Journal of Manufacturing Systems*, **15**(5), 351–366.

Montreuil, B. and LaForge, A. (1992) Dynamic layout design given a scenario tree of probable futures. *European Journal of Operational Research*, **63**(2), 271–286.

Montreuil, B. and Venkatadri, U. (1991) Strategic interpolative design of dynamic manufacturing systems layout. *Management Science*, **37**(6), 682–694.

Montreuil, B., Venkatadri, U. and Lefranois, P. (1991) Holographic layout of manufacturing systems. Technical report 91-76, Faculty of Management, Laval University, Quèbec, Canada.

Montreuil, B., Venkatadri, U. and Rardin, R.L. (1999) The fractal layout organization for job shop environments. *International Journal of Production Research*, **37**(3), 501–521.

Palekar, U.S., Batta, R., Bosch, R.M. and Elhence, S. (1992) Modeling uncertainties in plant layout problems. *European Journal of Operational Research*, **63**(2), 347–359.

Pardalos, P.M. and Wolkowicz, H. (1994) Quadratic Assignment and Related Problems. *DIMACS Series in Discrete Mathematics*, in American Mathematical Society.

Rosenblatt, M.J. (1986) The dynamics of plant layout. *Management Science*, **32**(1), 76–86.

Rosenblatt, M.J. and Kropp, D.H. (1992) The single period stochastic plant layout problem. *IIE Transactions*, **24**(2), 169–176.

Rosenblatt, M.J. and Lee, H.L. (1987) A robustness approach to facilities design. *International Journal of Production Research*, **25**(4), 479–486.

Shore, R.H. and Tompkins, J.A. (1980) Flexible facilities design. *AIIE Transactions*, **12**(2), 200–205.

Urban, T.L. (1992) Computational performance and efficiency of lower bound procedures for the dynamic facility layout problem. *European Journal of Operational Research*, **57**(2), 271–279.

Urban, T.L. (1993) A heuristic for the dynamic layout problem. *IIE Transactions*, **25**(4), 57–63.
Urban, T.L. (1998) Solution procedures for the dynamic facility layout problem. *Annals of Operations Research*, **76**, 323–342.
Urban, T.L., Chiang, W.C. and Russel, R.A. (2000) The integrated machine allocation and layout problem. *International Journal of Production Research*, **38**(13), 2911–2930.
Yang, T. and Peters, B.A. (1998) Flexible machine layout design for dynamic and uncertain production environments. *European Journal of Operational Research*, **108**(1), 49–64.

Appendixes

Appendix 1: The branch and bound algorithm

In this section, we present a branch and bound algorithm that optimally solves the distributed layout problem. For simplicity, we limit our discussion to the single-period problem. The algorithm could be extended to problems with multiple periods, although a solution to a non-trivial problem is unlikely for more than two periods.

In the algorithm, each branch of the branching tree represents a partial assignment of department copies. At each level of the tree, we branch on the assignment of all copies of a department to specific locations. For each branch, we calculate a lower bound on which we fathom non-promising branches: i.e., partial assignments that have a lower bound greater than the minimal cost of any complete assignment. We select the branch with the lowest available lower bound to perform the next branching. We use set I to denote the set of departments whose copies are assigned and set J to refer to the set of locations to which these departments are assigned. Once copies of department i are assigned to locations $k_1, \ldots, k_{N_i}$, department i is added to set I and the allocated locations are added to set J. For instance, at level $L = |I|$ of the branching tree, we have a partial assignment of all copies of L department types corresponding to a total of $\eta = (|\bar{J}| + 1)\sum_{i \in I} N_i + |J| \sum_{i \notin I} N_i$ fixed variables. The $\sum_{i \notin I} N_i$ unassigned department copies can still be assigned to any of the $\sum_{i \notin I} N_i$ unassigned locations. Given the assignment of all copies of department $i \in I$ to locations $k \in J$, we can calculate a lower bound for the objective function of the problem by splitting the problem into three independent subproblems (P1), (P2), and (P3) with objective functions z_1, z_2, and z_3 respectively:

1. Problem (P1) minimizes cost of flow between department copies that have been assigned locations;
2. Problem (P2) minimizes cost of flow between assigned and unassigned department copies; and
3. Problem (P3) minimizes cost of flow between unassigned department copies.

After substituting the values of the variables that correspond to assigned copies ($x_{nik} = 1$), problem (P1) (shown below) becomes a linear program, with flows v_{nimjp} as decision variables:

$$\text{(P1)}\quad \text{Min } z_1 = \sum_{p=1}^{P} \sum_{i \in I} \sum_{n=1}^{N_i} \sum_{j \in I} \sum_{m=1}^{N_j} \sum_{k \in J} \sum_{l \in J} v_{nimjp} x_{nik} x_{mjl} c_{kl} d_{kl}, \quad \text{(A1)}$$

subject to

$$\sum_{m=1}^{N_j} \sum_{n=1}^{N_i} v_{nimjp} = v_{ijp} \qquad \forall i \in I, j \in I, p, \qquad \text{(A2)}$$

$$\sum_{i \in I} \sum_{n=1}^{N_i} v_{nimjp} = \sum_{q \in I} \sum_{r=1}^{N_q} v_{mjrqp} \qquad \forall j \in I, m, p, \qquad \text{(A3)}$$

$$\sum_{p=1}^{P} \sum_{i \in I}^{N} \sum_{n=1}^{N_i} v_{nimjp} t_{mjp} \le C_{mj} \qquad \forall j \in I, m, \qquad \text{(A4)}$$

$$x_{nik} = 0, 1 \qquad \forall i \in I, n, k \in J, \qquad \text{(A5)}$$

$$v_{nimjp} \ge 0 \qquad \forall i \in I, n, j \in I, m, p. \qquad \text{(A6)}$$

Problem (P2) minimizes the cost of flow assignment between assigned and unassigned department copies. The problem is a restricted version of the original problem that is still not solvable in polynomial time:

$$\text{(P2) Min } z_2 = \sum_{p=1}^{P} \sum_{i \in I} \sum_{n=1}^{N_i} \sum_{j \notin I} \sum_{m=1}^{N_j} \sum_{k \in J} \sum_{l \notin J} (v_{nimjp} + v_{mjnip}) x_{nik} x_{mjl} c_{kl} d_{kl}, \qquad \text{(A7)}$$

subject to

$$\sum_{i \notin I} \sum_{n=1}^{N_i} x_{nik} = 1 \qquad \forall k \notin J, \qquad \text{(A8)}$$

$$\sum_{k \notin J} x_{nik} = 1 \qquad \forall i \notin I, n, \qquad \text{(A9)}$$

$$\sum_{m=1}^{N_j} \sum_{n=1}^{N_i} v_{nimjp} = v_{ijp} \qquad \forall i \in I, j \notin I, p, \qquad \text{(A10)}$$

$$\sum_{m=1}^{N_j} \sum_{n=1}^{N_i} v_{nimjp} = v_{ijp} \qquad \forall i \notin I, j \in I, p, \qquad \text{(A11)}$$

$$\sum_{i \in I} \sum_{n=1}^{N_i} v_{nimjp} = \sum_{q \in I} \sum_{r=1}^{N_q} v_{mjrqp} \qquad \forall j \notin I, m, p, \qquad \text{(A12)}$$

$$\sum_{i \notin I} \sum_{n=1}^{N_i} v_{nimjp} = \sum_{q \notin I} \sum_{r=1}^{N_q} v_{mjrqp} \qquad \forall j \in I, m, p, \qquad \text{(A13)}$$

$$\sum_{p=1}^{P} \sum_{i \in I} \sum_{n=1}^{N_i} v_{nimjp} t_{mjp} \le C_{mj} \qquad \forall j \notin I, m, \qquad \text{(A14)}$$

$$\sum_{p=1}^{P}\sum_{i\notin I}\sum_{n=1}^{N_i} v_{nimjp}t_{mjp} \le C_{mj}$$

$$-\sum_{p=1}^{P}\sum_{i\in I}^{N}\sum_{n=1}^{N_i} v_{nimjp}t_{mjp} \qquad \forall j \in I, m, \qquad \text{(A15)}$$

$$x_{nik} \in 0, 1 \qquad \forall i \notin I, n, k \notin J, \qquad \text{(A16)}$$

$$v_{nimjp} \ge 0 \qquad \forall i, n, j, m, p. \qquad \text{(A17)}$$

Problem (P3) is also a restricted version of the original problem, where only flows between unassigned department duplicates are considered.

$$\text{(P3) Min } z_3 = \sum_{p=1}^{P}\sum_{i\notin I}\sum_{n=1}^{N_i}\sum_{j\notin I}\sum_{m=1}^{N_j}\sum_{k\notin J}\sum_{l\notin J} v_{nimjp}x_{nik}x_{mjl}c_{kl}d_{kl}, \qquad \text{(A18)}$$

subject to

$$\sum_{i\notin I}\sum_{n=1}^{N_i} x_{nik} = 1 \qquad \forall k \notin J, \qquad \text{(A19)}$$

$$\sum_{k\notin J} x_{nik} = 1 \qquad \forall i \notin I, n, \qquad \text{(A20)}$$

$$\sum_{m=1}^{N_j}\sum_{n=1}^{N_i} v_{nimjp} = v_{ijp} \qquad \forall i \notin I, j \notin I, p, \qquad \text{(A21)}$$

$$\sum_{i\notin I}\sum_{n=1}^{N_i} v_{nimjp} = \sum_{q\notin I}\sum_{r=1}^{N_q} v_{nimjp} \qquad \forall j \notin I, m, p, \qquad \text{(A22)}$$

$$\sum_{p=1}^{P}\sum_{i\notin I}\sum_{n=1}^{N_i} v_{nimjp}t_{mjp} \le C_{mj}$$

$$-\sum_{p=1}^{P}\sum_{i\in I}\sum_{n=1}^{N_i} v_{nimjp}t_{mjp} \qquad \forall j \notin I, m, \qquad \text{(A23)}$$

$$x_{nik} = 0, 1 \qquad \forall i \notin I, n, k \notin J, \qquad \text{(A24)}$$

$$v_{nimjp} \ge 0 \qquad \forall i \notin I, n, j \notin I, m, p. \qquad \text{(A25)}$$

Whereas (P1) can be solved to optimality in polynomial time, there exists no polynomial-time algorithm to solve problems (P2) and (P3). A lower bound on z_2^* and z_3^* can be obtained by first reformulating problems (P2) and (P3) similarly to problem (PA) and relaxing the capacity constraints, and then applying the algorithm described in Section 4.

Appendix 2: Design of experiments for results in Figs. 6, 7, and 8

In order to examine the effect of product mix variability, systems with similar characteristics are generated, except that we consider two products per period, where we fix total demand per period (200 units per period for the examples shown in Fig. 6) but vary the contribution of each product in each period to reflect different levels of variability.

To study the effect of routing variability, we fix the number of department types to six types with four duplicates each. We conducted 45 sets of experiments with similar unit rearrangement costs ranging from one to 120, total demand per period varying from 200 to 400, and three levels of department duplication. To isolate the effect of routing variability, we consider layouts with two periods and one product per period. We fix the product routing (sequence of length seven) for the first period and adjust the routing for the second period according to the required routing-variability level. The results shown in Fig. 7 correspond to examples with 24 departments and a total demand per period of 300 units.

In order to examine the effect of volatility, we generate demand for a fixed set of products for a fixed set of time units (e.g., months). We then aggregate these time units into a set of periods, where a period would consist of one more time unit based on the desired level of volatility. Systems with the most volatility correspond to those where each time unit corresponds to an individual period. Intermediate levels of volatility are obtained by partitioning the time units into consecutive subsets of consecutive time units (e.g., quarters) and summing the demand of each product in these time units to obtain an aggregate demand for each product in each period (e.g., obtain quarterly demand for a product from its monthly demands). Systems with the least amount of volatility are those in which all the time units are aggregated into a single period. The results shown in Fig. 8 are for a system with eight time units and aggregate periods lasting one, two, four, and eight time units. Demand in each time unit for each product is randomly generated from a uniform distribution with a range of 100 to 200.

Biographies

Maher Lahmar is currently an Assistant Professor of Industrial Engineering at the University of Houston. He received B.S. and M.S. degrees in Industrial Engineering from Bilkent University, Ankara, Turkey, and a Ph.D. in Industrial Engineering from The University of Minnesota, Twin-Cities. His research interests are in the areas of manufacturing systems, production and inventory control, and facility planning. He is a member of IIE, INFORMS and SME.

Saif Benjaafar is a Professor in the Department of Mechanical Engineering at the University of Minnesota where he also serves as Director for the Center for Supply Chain Research and Director for the Industrial Engineering Division. He has been a Distinguished Senior Visiting Scientist with Honeywell Laboratories. He has also been a Visiting Professor at Ecole Centrale Paris and the Hong Kong University of Science and Technology. His research is in the area of supply chain management, production and inventory systems, and manufacturing and service operations. He serves as an Associate Editor for *IEEE Transactions on Automation Science and Engineering, IIE Transactions*, and *International Journal of Flexible Manufacturing Systems*. He holds Ph.D. and M.S. degrees in Industrial Engineering from Purdue University and a B.S. degree in Electrical Engineering from the University of Texas at Austin.

Contributed by the Focused Issue Editor