TECHNICAL RESEARCH REPORT

Design of Material Flow Networks in Manufacturing Facilities

by J.W. Herrmann, G. Ioannou, I. Minis R. Nagi, and J.M. Proth

T.R. 94-50

INSTITUTE FOR SYSTEMS RESEARCH

*Sponsored by
the National Science Foundation
Engineering Research Center Program,
the University of Maryland,
Harvard University,
and Industry*

Design of Material Flow Networks in Manufacturing Facilities

J.W. Herrmann* G. Ioannou† I. Minis† R. Nagi‡ J.M. Proth§

Abstract

In this paper we consider the design of material handling flow paths in a discrete parts manufacturing facility. A fixed-charge capacitated network design model is presented and two efficient heuristics are proposed to determine near-optimal solutions to the resulting *NP*-hard problem. The heuristics are tested against an implicit enumeration scheme used to obtain optimal solutions for small examples. For more realistic cases, the solutions of the heuristics are compared to lower bounds obtained by either the linear programming relaxation of the mixed integer program, or an iterative dual ascent algorithm. The results obtained indicate that the heuristics provide good solutions in reasonable time on the average. The proposed methodology is applied to design the flow paths of an existing manufacturing facility. The role of the flow path network problem in the integrated shop design is also discussed.

*Institute for Systems Research, University of Maryland, College Park, MD 20742.

†Department of Mechanical Engineering and Institute for Systems Research, University of Maryland, College Park, MD 20742.

‡Department of Industrial Engineering, SUNY at Buffalo, Buffalo, NY 14620

§INRIA-Lorraine, 4 Rue Marconi, Metz 2000, 57070 Metz, France.

1 Introduction

The overall performance of an industrial firm may be significantly impacted by the design of its manufacturing facility. A well-designed facility results in efficient material handling, small transportation times, and short queues. This in turn leads to low work-in-process, effective production management, decreased cycle times and manufacturing inventory costs, improved on-time delivery performance, and, consequently, higher product quality [16, 19]. The global problem of facility design includes three interrelated tasks: i) placing the manufacturing resources (machines, departments or cells) within the available area of the shop (shop layout), ii) selecting the material handling system (MHS) flow paths or aisles, and iii) designing the MHS, e.g. determining the number of transporters necessary to serve the material flow as well as the assignment of empty transporter moves.

A large body of literature has focused on the layout problem, and several powerful methods have been developed to place the shop resources in such a manner that inter-resource material handling is minimized [6, 7]. But although the material handling effort also depends upon the topology of the network connecting the resources, limited attention has been paid to developing good network designs.

The research in MHS flow path design has focused on a few types of MHS, such as Automated Guided Vehicle Systems (AGVS) [23]. For the case of unidirectional AGVS with fixed aisles, the problem of determining the flow path directions was first formulated by Gaskins and Tanchoco [4] as a zero-one integer programming problem. This model has been extended by Kaspi and Tanchoco [10] and Sinriech and Tanchoco [21]. Goetz and Egbelu [5] developed a heuristic approach that reduces the number of constraints required. For the same type of MHS, Venkataramanan and Wilson [22] presented a similar but more compact formulation based on strongly connected graphs. They also extended the model to account for unloaded flow information. All these models were solved by different branch-and-bound algorithms [10, 11, 21, 22]. Sharp and Liu [20] developed an analytical method for configuring fixed-path, closed-loop MHS based on a mixed integer programming formulation. Egbelu and Tanchoco [3] also studied the merits of bidirectional AGVS. Their simulation results showed that the efficiency and productivity of the manufacturing shop is increased, compared to unidirectional AGVS, at the cost of control complexity and considerable guide path investment.

All studies mentioned above are limited in scope, since they address a special fixed aisle system. In this case, the only design variable is the direction of each edge in the associated graph, the configuration of which has been fixed *a priori*. Kim and Tanchoco [12] proposed a network design model which accounts for both transportation costs and fixed costs such as construction, space and control costs. Their solution approach consists of an enhanced branch-and-bound approach that employs a tighter bound and a more efficient search scheme compared to the one in [10].

Beyond AGVS, a few research studies have considered general MHS flow path design problems. Proth and Souilah [19] proposed a fast branch-and-bound algorithm to evaluate the shortest path between two departments/cells, which may serve as the flow path of the MHS. Their method, though applicable to every type of horizontal transportation system, does not account for practical system constraints, such as material flow bounds within MHS aisles to prevent congestion. Montreuil and Ratliff [18] suggested a cut tree algorithm to generate a minimum weight spanning tree, the edges of which represent flow path segments. Weights which reflect the minimum cut tree flows are assigned to each edge. Based on these weights, the flow path that corresponds to the minimum material handling cost is determined. This is accomplished by adjusting the edge lengths in order to minimize the cumulative product of the flow through each edge and the edge length. The cut tree method is valid only when the flow path is *a priori* selected to be a spanning tree of the graph of the resource input/output stations. Thus, it cannot address closed loops or multi-row configurations, which are usually preferable. Chhajed *et al.* [2] impose a grid on the entire facility, the edges of which can be used as MHS aisles. The MHS flow path design consists of selecting the most appropriate edges of the grid and is formulated as a mixed integer problem. A Lagrangian relaxation of the mixed integer formulation decomposes the problem into shortest path subproblems that may be solved in linear time. However, their formulation allows flow paths to pass through entity-occupied areas, which is clearly impractical. Finally, Maxwell and Wilson [15] developed a network flow model for analyzing the traffic in dynamically loaded MHS with fixed paths. Their method is an analytical tool that can be used to evaluate the performance of candidate designs.

In this paper, we concentrate on the design of material flow networks for horizontal MHS, including automated guided vehicles, manual or automated rail carts, industrial trucks, and forklifts. Under certain conditions (i.e. if the handling operations follow the aisle network), overhead cranes and bi-directional conveyors, can be also included. A fixed-charge capacitated network design model is introduced, which incorporates critical practical concerns such as fixed costs, operating costs, and aisle capacities. This network design problem is *NP*-hard [9]. We propose two heuristic methods which generate near-optimal solutions to realistic-sized problems. The quality of the solutions to small problems is assessed by comparing them to the optimal solutions derived by a branch-and-bound procedure. The solutions to larger problems are compared to lower bounds obtained through a linear relaxation of the mixed integer program and through a dual ascent approach. The results indicate that both heuristics provide good solutions in reasonable time for real life problems. Since this design problem takes place at the planning stage of a manufacturing system, computation time is not a key objective, assuming that it remains at a reasonable level. Thus the proposed model and the two heuristics can be employed to design efficient material flow networks in manufacturing facilities.

The paper is organized as follows: In Section 2 the flow path design problem is formulated as a multi-commodity fixed-charge capacitated network design model. In Section 3 two heuristic procedures are proposed to determine near-optimal solutions. In Section 4 these heuristics are applied to several shop design examples in order to assess the quality of the solutions obtained. In addition, Section 4 presents an industrial application of the design methodology. Finally, Section 5 discusses the applicability of the methodology to the global shop design problem and summarizes the conclusions of this study.

2 Problem Formulation

Consider an orthogonal unit grid imposed on the area of the manufacturing shop (see Figure 1). The unit length of the grid is defined such that it is larger than the width of a typical MHS aisle and it is fine enough to adequately capture the geometry of the shop, including restricted areas, and the geometry of the manufacturing departments/cells. Each intersection of the grid represents a node of the underlying graph (i.e. the graph from which the material flow network will be constructed). Note that grid intersections which are inside restricted areas or areas occupied by manufacturing entities (e.g. point A in Figure 1) are not graph nodes. The only exceptions are special nodes that coincide with input and output stations of cells or departments (through which material enters and leaves the department/cell, respectively). Two such nodes are denoted by I and O in Figure 1.

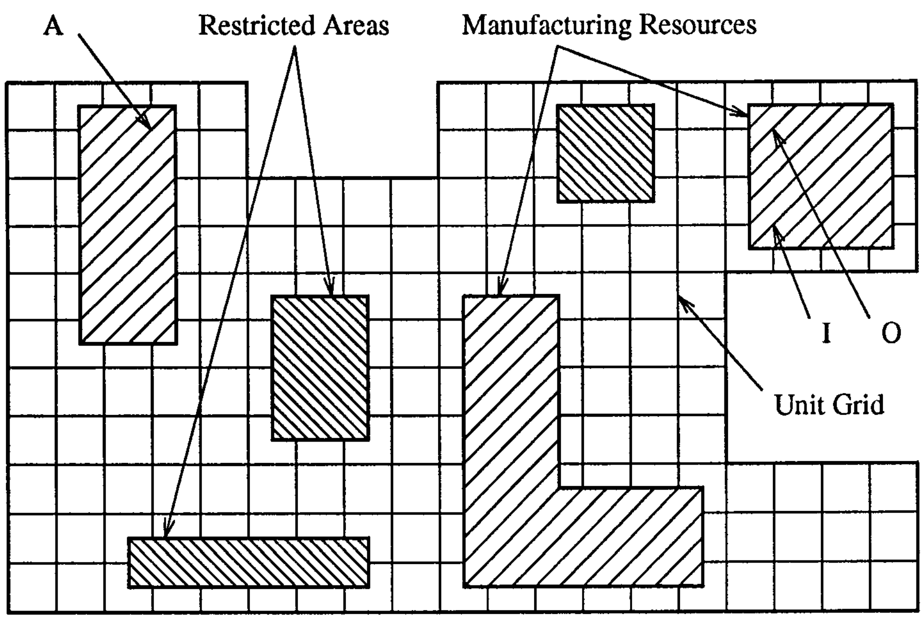

Figure 1: Manufacturing shop and department/cell representations

Let N be the set of grid nodes, $\bar{A}$ the set of undirected arcs $\{i, j\}$ that connect these nodes, and $G = (N, \bar{A})$ the associated graph. Note that all arcs have unit lengths, since they connect two adjacent nodes. Also, since the flow is directed, let A be the set of directed arcs that correspond to $\bar{A}$, i.e. $(i, j), (j, i) \in A$ if $\{i, j\} \in \bar{A}$. We consider the problem of selecting some of the arcs of this graph to form the material flow network. Two important cost contributors are addressed:

1. Fixed Cost: Including an arc in the material flow network results in a certain fixed charge. This accounts for the cost of constructing this segment of the flow path (e.g. the unit cost of installing a conveyor or an AGVS), the space cost (since devoting a part

of the shop to the MHS makes that area otherwise unavailable) and the corresponding control cost (e.g. traffic control at path intersections).

2. Variable Cost: This represents the cost of routing parts across the network, i.e. the material handling cost.

These costs are in conflict, since including a large number of arcs may offer a substantial reduction in the routing (variable) costs due to shorter origin-destination paths at the expense of increased fixed charges. On the other hand, using fewer arcs in the final design results in lower fixed costs and higher routing costs (due to longer origin-destination paths). Thus, the objective is to achieve the best tradeoff between the fixed and variable costs of the network; i.e. to provide efficient material handling through a flow network that is inexpensive to construct and operate.

This problem falls into the category of fixed-charge network design models, which arise in a variety of problems and have attracted significant research interest. Magnanti and Wong [14] and Minoux [17] survey alternative formulations, recent solution approaches, and applications of this problem.

Let K be the set of material flows, or commodities, between the resources (departments or cells); i.e. the part traffic from resource d_a to resource d_b is represented by a single commodity $k \in K$. The flow intensity between d_a and d_b, or the flow for commodity k within a certain time horizon, is denoted by f_k. For each commodity k, let $O(k)$ be its origin, i.e. the output station of resource d_a, and $D(k)$ its destination, i.e. the input station of resource d_b. In addition, let F_{ij} be the fixed-charge associated with arc $\{i,j\} \in \bar{A}$. This reflects the construction, space, and control costs discussed above. Note that different fixed costs can be assigned to various arcs in order to favor certain network configurations. Furthermore, let B_{ij} be the capacity of arc $\{i,j\} \in \bar{A}$, i.e. the traffic that this arc can accommodate within the design time horizon. Finally, let c_{ij}^k be the routing cost for commodity $k \in K$ on arc (i,j). This may reflect the length of the arc (if different from unity) or other design considerations and may be different for each commodity. The routing costs are generally symmetrical, i.e. $c_{ij}^k = c_{ji}^k$.

Two types of variables are required for the problem formulation. The first type comprises continuous variables x_{ij}^k that represent the fraction of the k-th commodity flow that travels across arc $(i,j) \in A$. The second type comprises binary variables y_{ij} that model discrete choices as follows:

$$y_{ij} = \begin{cases} 1 & \text{if arc } \{i,j\} \text{ is chosen as part of the network design} \\ 0 & \text{otherwise} \end{cases}$$

Also, let $N(i)$ be the set of nodes adjacent to node $i \in N$.

Based on these conventions, the flow path design problem can be formulated as a multi-commodity fixed charge capacitated network design model as follows:

Problem $P(G)$

$$\text{minimize} \qquad Z = \sum_{k \in K} f_k \sum_{(i,j) \in A} c_{ij}^k x_{ij}^k + \sum_{\{i,j\} \in \bar{A}} F_{ij} y_{ij} \tag{1}$$

subject to :

$$\sum_{j \in N(i)} x_{ji}^k - \sum_{l \in N(i)} x_{il}^k = \begin{cases} -1 & \text{if } i = O(k) \\ 1 & \text{if } i = D(k) \quad \forall i \in N, k \in K \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$\sum_{k \in K} f_k (x_{ij}^k + x_{ji}^k) \le B_{ij} \qquad \forall \{i,j\} \in \bar{A} \tag{3}$$

$$x_{ij}^k, x_{ji}^k \le y_{ij} \qquad \forall \{i,j\} \in \bar{A}, k \in K \tag{4}$$

$$x_{ij}^k \ge 0 \qquad \forall (i,j) \in A, k \in K \tag{5}$$

$$y_{ij} \in \{0,1\} \qquad \forall \{i,j\} \in \bar{A} \tag{6}$$

The objective function Z reflects the basic tradeoff between the routing costs and the fixed costs for using network arcs. It accounts for the material flow within the shop as well as the fixed cost for building a particular flow network. Constraints (2) are the flow conservation equations imposed on each flow $k \in K$. They ensure the continuity of the flow path between each origin-destination pair. Constraint set (3) is critical for the effective operation of the material handling system. It limits the flow through an arc to a bound, B_{ij}, in order to prevent traffic congestion or to account for the capacity of the MHS on that arc. Note that including additional flow path segments will be necessary to accommodate flow beyond an arc's capacity. This is also consistent with vehicle collision avoidance when alternative routes are followed, to avoid MHS conflicts (see Krishnamurty *et al.* [13]). The value of B_{ij}

depends upon the number of material handling units in the system, the overall flow intensity (both loaded and unloaded), and the traffic control at path intersections. Constraint set (4) prohibits flow through non-selected arcs, i.e. arcs $\{i, j\} \in \bar{A}$ with $y_{ij} = 0$. Finally, constraints (5) ensure the non-negativity of the continuous variables x_{ij}^k, while constraints (6) force the variables y_{ij} to assume binary values.

Since the flow path design problem is intrinsically coupled with the shop layout problem it is important for these two problems to address positively correlated, if not identical, objectives. The following remark shows that objective function (1) incorporates the typical objective of shop layout optimization models, i.e. the product of material flow between resources and the associated distance, summed over all pairs of resources.

Remark 1 *If $c_{ij}^k = 1$, $\forall (i, j) \in A, k \in K$ and G comprises only unit length arcs, then the first term of the minimization objective (1) is the cumulative material flow times distance.*

Note that since for each commodity the first summation in (1) represents the flow times the distance, the sum over all commodities is the overall material handling cost.

3 Solution Approach

As mentioned previously, the multi-commodity fixed charge capacitated network design problem is *NP*-hard. Implicit enumeration schemes can provide the optimal solution for only small problems. Consequently, we concentrate on the development of heuristics in order to obtain near-optimal solutions, and the construction of lower bounds to assess solution quality. In this section, we present two heuristic methods that generate a feasible solution to the mixed integer program of $P(G)$. They both take advantage of the linear programming relaxation of problem $P(G)$ and proceed by iteratively fixing one or more of the y_{ij} variables to 1 or 0.

3.1 A fixed-charge adjustment heuristic (FCAH)

Let $P'(G)$ refer to the linear programming relaxation of problem $P(G)$, i.e. the linear program derived from $P(G)$ by replacing the binary variables $y_{ij} \in \{0, 1\}$ by continuous variables $y_{ij} \in [0, 1]$. Also, let (x, y) be an ordered pair, where x is the vector of real variables x_{ij}^k and y the vector of continuous variables $y_{ij} \in [0,1]$ of the relaxed problem, which correspond to the binary variables of $P(G)$. Suppose (x^o, y^o) is an optimal solution to $P'(G)$.

The FCAH algorithm is based on the assertion that elements of y^o with values close to one ($y_{ij}^o \geq 1-\varepsilon$, where $\varepsilon \approx 0$) represent strong candidates for inclusion in the active network. Similarly, the elements of y^o with values close to zero ($y_{ij}^o \leq \varepsilon$) represent strong candidates for exclusion. Consequently, a good feasible integer solution may be reached by favoring the first class of arcs and penalizing the second. This is accomplished by adjusting the fixed charge values F_{ij}.

Let $\varepsilon \approx 0$ be a tolerance factor, $r > 1$, a correction factor, and $p_1 > p_2 > 1$ two integer parameters that control the length of the search. The steps of the proposed heuristic are as follows:

Algorithm FCAH$(\varepsilon, p_1, p_2, r)$

Step 1: Initialization

$p = 1$, $q = 1$

$A_q = A$, $\bar{A}_q = \bar{A}$, $S = \bar{A}$

$G_q = (N, \bar{A}_q)$

Step 2: Solution of linear programming relaxation

If $S = \emptyset$ then go to *Step 7*

Solve linear program $P'(G_q)$ to obtain (x^o, y^o)

if $y_{ij}^o \in \{0,1\}$ $\forall \{i,j\} \in \bar{A}_q$ then go to *Step 7*

Step 3: Variable selection

Select arcs $\{s,t\} \in \bar{A}_q$ and $\{u,v\} \in \bar{A}_q$ such that

$$y_{st}^o = max\{y_{ij}^o : \{i,j\} \in \bar{A}_q, y_{ij}^o \neq 1\}$$

$$y_{uv}^o = min\{y_{ij}^o : \{i,j\} \in \bar{A}_q, y_{ij}^o \neq 0\}$$

Step 4: Fixed-charge update

$F_{st} \leftarrow F_{st}/r$

$F_{uv} \leftarrow F_{uv} \cdot r$

Step 5: Fix binary variables; update arc set if $q < p_2$ then

$\forall\{i,j\} \in \bar{A}_q : y^o_{ij} \geq 1 - \varepsilon$

$y^o_{ij} = 1$

Add constraint $y_{ij} = 1$ to $P(G_q)$ and any subsequent program

$S \leftarrow S \setminus \{i,j\}$,

$\forall\{i,j\} \in \bar{A}_q : y^o_{ij} \leq \varepsilon$

$y^o_{ij} = 0$

$\bar{A}_1 \leftarrow \bar{A}_q \setminus \{i,j\}$, $A_1 \leftarrow A_q \setminus \{(i,j),(j,i)\}$

$S \leftarrow S \setminus \{i,j\}$,

if $\exists\{i,j\} \in \bar{A}_q : (y^o_{ij} = 1)$ or $(y^o_{ij} = 0)$

$q = 1$ and go to *Step 6*

else

$\bar{A}_{q+1} \leftarrow \bar{A}_q$, $A_{q+1} \leftarrow A_q$, $G_q = (N, A_q)$

$q \leftarrow q + 1$ and return to *Step 2*

else

Add constraint $y_{st} = 1$ to $P(G_q)$ and any subsequent program

$\bar{A}_{q+1} \leftarrow \bar{A}_q \setminus \{s,t\}$, $A_{q+1} \leftarrow A_q \setminus \{(s,t),(t,s)\}$

$G_q = (N, A_q)$

$q \leftarrow q + 1$ and return to *Step 2*

Step 6: Final variable update

if $p > p_1$ then

$y^o_{ij} = 1\ \forall\{i,j\} \in \bar{A}_{q+1}$ and go to *Step 7*

else

$p \leftarrow p + 1$ and return to *Step 2*

Step 7: End

output near-optimal solution (x^o, y^o) and objective function value Z

(with respect to the original fixed costs)

Let us now discuss the parameters and the steps of the algorithm. The intent is to force the variables y_{ij} to either zero or one. Whenever the value of any y_{ij}, which is computed by the linear relaxation of $P(G_q)$, is within a tolerance factor ε from either 0 or 1, the variable is fixed and the associated arc is either removed from the graph or activated. Then the problem is reformulated accordingly. The role of the tolerance factor is obvious. A smaller value of ε yields a more accurate prediction of the value of the variables in the optimal solution; in this case, however, more computational effort is required in order to fix the values of the variables. On the other hand, if ε is large the algorithm may converge quickly to a poor solution, or it may even fail to converge to a feasible solution. That is why ε should be less than the smallest possible value of all the x_{ij}^k variables.

The performance of the algorithm is very sensitive to the correction factor r. The role of r is similar to the idea of Lagrangian multipliers or to the penalty methods for constrained optimization. Step 3 identifies the y_{ij} closest to 1 in the solution of the linear relaxation of problem $P(G_q)$ (Step 2). By multiplying the fixed charge of arc $\{i, j\}$ by a factor of $1/r$, the probability that this arc will be included in the solution is increased. Consequently, in the next iteration increased flow will be routed through arc $\{i, j\}$, and the linear programming relaxation will absorb in the objective function more of the arc's fixed charge (since $x_{ij}^k \leq y_{ij}$). As a result, y_{ij} may be within ε of 1 when the relaxed problem is re-solved. The reverse occurs for those y_{ij} which are close to zero. The fixed charge is multiplied by r to make the arc undesirable and drive y_{ij} to zero. The selection of r should take into account the following extreme cases : A large correction factor ($r \gg 1$) will lead to a myopic local optimum, while a small correction factor ($r \approx 1$) will substantially increase the time necessary for the heuristic to converge to a binary solution.

The maximum number of global iterations allowed is specified by parameter p_1. In case the number of iterations reaches p_1, the variable fixing procedure of Step 6 is invoked. The value of p_1 is chosen to achieve the best tradeoff between computation time and solution quality. Parameter p_2 controls the number of loops within each iteration allowed without updating the set of arcs, the binary values of which are not fixed. It is clear that increasing p_2 increases the computational effort of the algorithm.

Remark 2 *Algorithm FCAH converges to a feasible solution, if one exists.*

The convergence of FCAH is guaranteed by Step 6. FCAH would fail to converge only if some necessary arc to a feasible solution was excluded in Step 5. This exclusion may occur if $\varepsilon \geq x_{ij}^k$ for some $(i,j) \in A$ and $k \in K$, so ε must be chosen less than the smallest possible value of the x_{ij}^k variables. Otherwise, the feasibility of the linear relaxations at each iteration guarantees that a feasible network can be constructed by rounding up the fractional y_{ij}^o.

3.2 A depth-first-search algorithm

Consider the state space of the mixed-integer program of Eqs.(1)-(6). By state space we refer to the set of combinations of the values of the binary variables. If problem $P(G)$ has $n = |\bar{A}|$ variables, the state space consists of 2^n distinct points. It is obvious that for medium to large problems it is not possible to evaluate the objective function value for each of these states. Thus, a heuristic that partially explores the binary tree (a tree representation of the state space of the problem) is proposed.

In this heuristic, the state space tree of the mixed-integer program $P(G)$ is explored until a feasible integer solution is obtained; i.e. the search terminates when a leaf of the tree is encountered. This type of search from the root of the tree towards a leaf is called a depth-first search. However, the search implemented here is enhanced to explore the most promising nodes first. These nodes are associated with variables y_{ij}, the values of which are closer to either zero or one in the solution of the linear program $P'(G)$. The basic tool employed to evaluate the lower bound at each tree node is the linear programming relaxation of problem $P(G)$. The Ordered Depth-First Search (ODFS) algorithm is as follows:

Algorithm ODFS

Step 1: Initialization

Solve linear program $P'(G)$ to obtain (x^o, y^o)

Rank variables y_{ij} is descending order of $|y_{ij}^o - 0.5|$

Store the y_{ij}-variables in list L in the above order

Step 2: Depth-first-search

Set y_{pq} = first element of L and remove it from the list

Set $G_o = (N, A \setminus \{p,q\})$ and $y_{pq} = 0$

Set $G_1 = (N, A)$ and $y_{pq} = 1$

Step 3: Solution of the linear programming relaxations

Solve $P'(G_o)$ to obtain the objective function value Z_o

Solve $P'(G_1)$ to obtain the objective function value Z_1

Step 3: Variable fixing

If $Z_o \leq Z_1$ then set $y^o_{pq} = 0$ and $A \leftarrow A \setminus \{p, q\}$

Else set $y^o_{pq} = 1$

Step 4: Check for variables not yet fixed

If $L \neq \emptyset$ return to *Step 2*

Step 7: End

Output near-optimal solution (x^o, y^o) and objective function value Z

The above depth-first search algorithm is straight forward. In Step 1, the values of the variables y_{ij} are derived by solving the linear programming relaxation of $P(G)$. Then, the variables y_{ij} are ranked with respect to their distance from 0.5. Note that this procedure will rank at the top of the list L those y_{ij} variables with values closer to either zero or one. These variables are fixed at the early stages of the search. Subsequently, the depth-first search procedure is invoked. In Step 2, the first element y_{pq} of the list L is selected and the corresponding arc is either included to or excluded from the graph. This is accomplished by solving the resulting linear programming relaxations in Step 3 and fixing the variable y_{pq} to the value that yields the best lower bound. The procedure is repeated until all variables are fixed.

The effectiveness of ODFS is directly related to the quality of the linear programming approximation of the mixed-integer program.

3.3 Exact methods and lower bounds

A straightforward branch-and-bound scheme is employed to assess the effectiveness of the heuristics in solving small problems. It extends the ordered depth-first search heuristic and explores implicitly the whole binary tree. Standard branch pruning and lower, as well as upper, bound update techniques are implemented. Since each node of the tree requires the solution of two linear programs, the computational time is significantly large. For our computer implementation, only problems that comprise 30 or fewer arcs were solved to optimality.

In order to evaluate the quality of the heuristic solution procedures for larger problems, good lower bounds are required. Since the linear programming relaxation of the mixed integer program is not an adequate approximation to problem $P(G)$, it does not, in general, provide a good lower bound. In this work, the dual ascent algorithm of Herrmann *et al.* [8] is used to obtain good lower bounds. This algorithm, which is based on the labeling method of Balakrishnan *et al.* [1], is presented in Appendix A and, in general, provides lower bounds better than those of the linear programming relaxation. Both lower bounds are employed for the evaluation and comparison of the heuristics.

4 Numerical Results

The quality of the solutions derived by the proposed heuristics is assessed in this section through extensive computational experiments on various graph, origin-destination, and material flow patterns. Also, the methodology is applied to the design of the flow paths in an existing manufacturing shop.

4.1 Evaluation of the Heuristics

Both heuristics (FCAH and ODFS) were implemented in C on a Sun Sparc IPX station. Numerous computational tests were performed. In this section we present the results of solving randomly generated case problems with 20 to 60 arcs. These are problems on grid graphs, a network configuration that complies with the graph representation of the manufacturing facility. In each case, 50 problem instances were solved. The parameters employed to generate the case problems were uniformly distributed over the ranges shown in Table 1.

In Table 2 the solutions of the fixed cost adjustment heuristic (FCAH) are compared to the optimum (for the 20 and 30 arc cases) as well as the lower bounds obtained from the dual ascent procedure and the linear relaxation, respectively (for all cases). The first column of Table 2 lists the average percentage difference between the solution of FCAH (Z_{FCAH}) and the optimum (Z_{opt}). The second column lists the average percentage difference between

Table 1: Ranges of randomly selected parameters in case problems

Parameter	Range
Fixed charge	5 - 40
Number of commodities	10 - 35
Commodity flow	4 - 12
Arc capacity	30 - 250

Z_{FCAH} and the bound of the iterative dual ascent algorithm (Z_{DA}). Similarly, the third column lists the average percentage difference of Z_{FCAH} from the linear relaxation bound (Z_{LP}).

Table 2: Comparison of FCAH solution vs. optimum and lower bounds

(50 problems per case)

	Optimum	Dual ascent	Linear relaxation
Number of arcs	$\frac{Z_{FCAH}-Z_{opt}}{Z_{opt}} \times 100\%$	$\frac{Z_{FCAH}-Z_{DA}}{Z_{DA}} \times 100\%$	$\frac{Z_{FCAH}-Z_{LP}}{Z_{LP}} \times 100\%$
20	2.4	4.5	8.6
30	3.2	6.3	9.4
40	-	7.9	10.3
50	-	7.4	9.6
60	-	6.8	9.2

As expected, for small examples and depending on the structure of the network, the heuristic may perform extremely well, since the number of alternative solutions is relatively small. This is especially true if the number of flows is large, in which case most nodes of the graph are either origins or destinations of flows. For larger problem instances, the heuristic solution may diverge from the optimal; however, as the problem size increases, its performance on the average is expected to improve. Finally, Table 2 shows that Z_{DA} is always better than Z_{LP} (see also [8]).

Table 3 summarizes the results of the performance tests for the ordered depth-first search scheme (Z_{ODFS}). Its performance is slightly inferior to the fixed cost adjustment heuristic.

In addition, FCAH required significantly less computational effort, especially when the convergence parameters were tuned well.

Table 3: Comparison of ODFS solution vs. optimum and lower bounds

(50 problems per case)

Number of arcs	Optimum $\frac{Z_{ODFS}-Z_{opt}}{Z_{opt}} \times 100\%$	Dual ascent $\frac{Z_{ODFS}-Z_{DA}}{Z_{DA}} \times 100\%$	Linear relaxation $\frac{Z_{ODFS}-Z_{LP}}{Z_{LP}} \times 100\%$
20	2.7	4.8	8.9
30	3.6	6.4	9.6
40	-	8.2	10.4
50	-	7.6	9.7
60	-	7.2	9.5

From Tables 2 and 3 we can see that the average difference between the best solution derived by either heuristics and the average lower bound of the dual ascent is less than 8% in all problem instances.

Table 4: Divergence measures

	cases with solutions within 10% of Z_{DA}	cases with solutions 20% or more greater than Z_{DA}
FCAH	243	0
ODFS	226	8

It is worth noting that FCAH provided good solutions consistently, while ODFS resulted occasionally in very poor solutions. Table 4 shows the number of problems (out of 250) for which the two heuristics resulted in solutions that are within 10% of the dual ascent lower bound Z_{DA}. It is clear that most of the FCAH solutions are included in this range, while almost 10% of the ODFS solutions are not. Furthermore, there were some cases for which the ODFS provided solutions which are 20% or more away from Z_{DA} (see Table 4).

4.2 Industrial Application

In order to illustrate the application of the proposed method the shop of a small-size manufacturer of vertical blinds is considered. The shop comprises eighteen resources, numbered consecutively from 1 to 18, which perform all operations from receiving the raw material to shipping the finished products. It has dimensions $960' \times 600'$ and is dedicated to the manufacture of five types of final assemblies, including wood and plastic blinds.

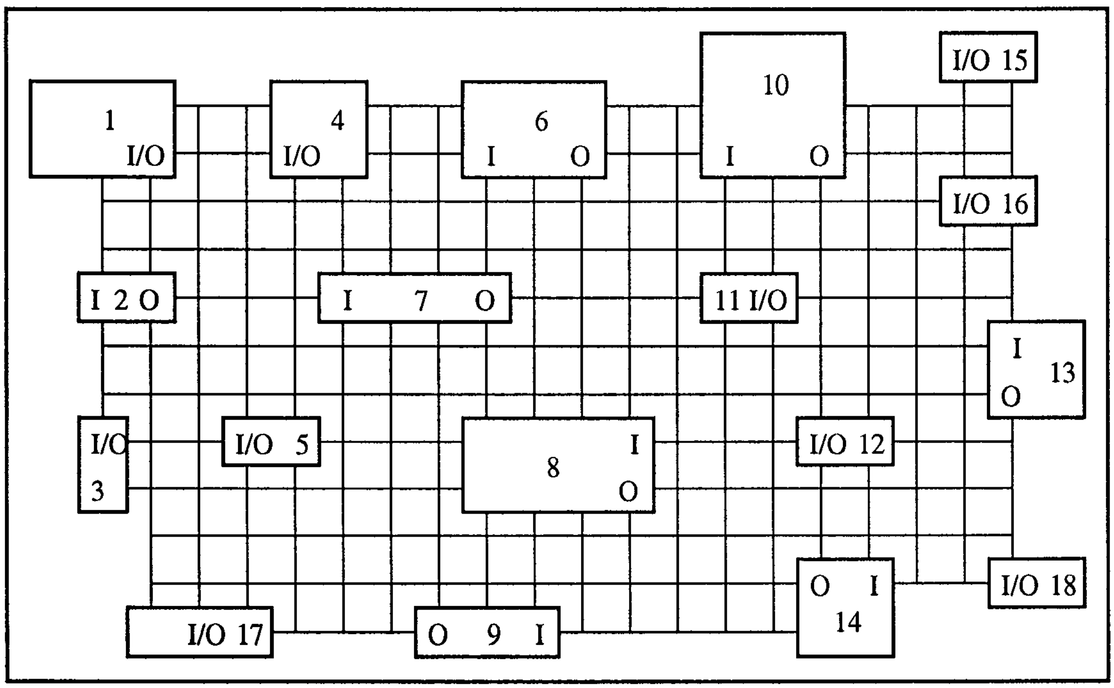

Figure 2: Sample manufacturing shop and cells

Figure 2 shows a model of the existing shop and the grid imposed on the shop floor. Each intersection of the grid is a node of the set N, while each horizontal and vertical segment is a candidate network arc (in the set $\bar{A}$). The locations of the input (I) and output (O) stations for each cell are also shown in Figure 2.

The production rates vary from 12 to 126 units per shift, depending on the product type and the forecasted demand for the next two years. Several make parts are required for each final assembly.

The material flow (from-to traffic) matrix of all make parts is given in Table 5. The entries of this table correspond to the demand per shift, which is derived by the forecasted demand for the five final assemblies over a two year period. Each entry a_{ij} represents the number of parts which must be transported from the output station of resource d_i to the input station of resource d_j.

Table 5: From-to flow matrix

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	.	.	83	.	195	.	.	.	.	.	.	.	.	.	.	.	.	.
2	.	.	.	293	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	.	.	.	.	51	.	.	.	.	.	.	.	.	.	.	32	.
4	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	293	.
5	.	.	.	.	.	.	.	195	.	.	.	.	.	.	.	.	.	.
6	.	.	.	.	.	.	.	.	.	51	.	.	.	.	.	.	.	.
7	.	.	.	.	.	.	.	.	.	195	.	.	.	.	.	.	.	.
8	.	.	.	.	.	.	195	.	.	.	.	.	.	.	.	.	.	.
9	.	.	.	.	.	.	.	.	.	.	.	.	.	325	.	.	.	.
10	.	.	.	.	.	.	.	.	.	.	.	.	.	.	51	195	.	.
11	.	.	.	.	.	.	.	.	.	.	.	325	.	.	.	.	.	.
12	.	.	.	.	.	.	.	.	325	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	.	.	325	.	.	.	.	.	.	.
14	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	325
15	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	325	.	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

Three cases have been considered for the flow path design problem of this shop (cases W_1, W_2 and W_3). All three are based on the capacitated fixed-charge network design formulation. In case W_1, the fixed charges are small and the capacities are large. This case corresponds to the design of the flow paths for an inexpensive horizontal transportation system, such as manually operated carts, which will use many flow paths to efficiently move material between resources. A fixed charge of $F_{ij} = 100$ units has been assigned to each candidate arc, and for every commodity the variable cost was set equal to the scaled arc length (1 unit). The arc capacities were fixed to $B_{ij} = 1500$ units of flow. The best configuration was obtained by FCAH and is shown in Figure 3. This solution corresponds to a total cost $Z = 46871$; i.e. fixed cost $Z_f = 10800$ and variable cost $Z_v = 36071$.

In this design, all commodities are routed through shortest paths. Although alternative shortest paths exist for several origin-destination pairs, the presence of fixed charges has

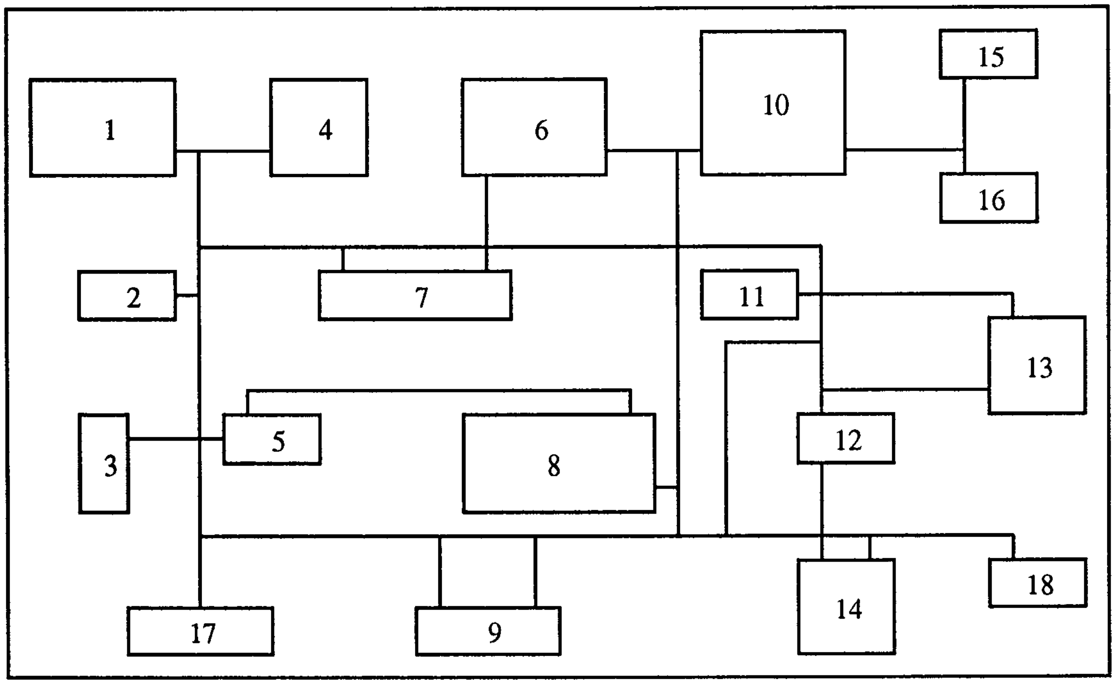

Figure 3: Inter-resource material flow paths (W_1)

forced commodities to employ common network arcs, thus providing a simpler and more attractive flow path solution. Furthermore, the capacity constraints were not active in the heuristic solution.

The second case examined (case W_2) is the capacitated network design with large fixed charges and inactive capacities. This situation may occur when the installation of expensive high volume material handling equipment is the primary consideration. In this case, a fixed charge of $F_{ij} = 500$ units has been assigned to each candidate arc (as in case W_2), while the capacity was set to $B_{ij} = 3779(= \sum_{i=1}^{18} \sum_{j=1}^{18} a_{ij})$ units of flow, to guarantee that capacity constraints are inactive. For every commodity the variable cost was set equal to the scaled arc length (1 unit). Figure 4 shows the network design obtained by the FCAH for this case. The total cost is $Z = 82341$; i.e. fixed cost $Z_f = 42500$ and variable cost $Z_v = 39841$.

It is clear that the configuration of Figure 4 has a smaller number of active arcs compared to case W_1), as a result of the substantial fixed charge. In addition, the topology of the flow network is very simple due to the absence of capacity constraints.

The third case considered (case W_3) is the capacitated network design with large fixed

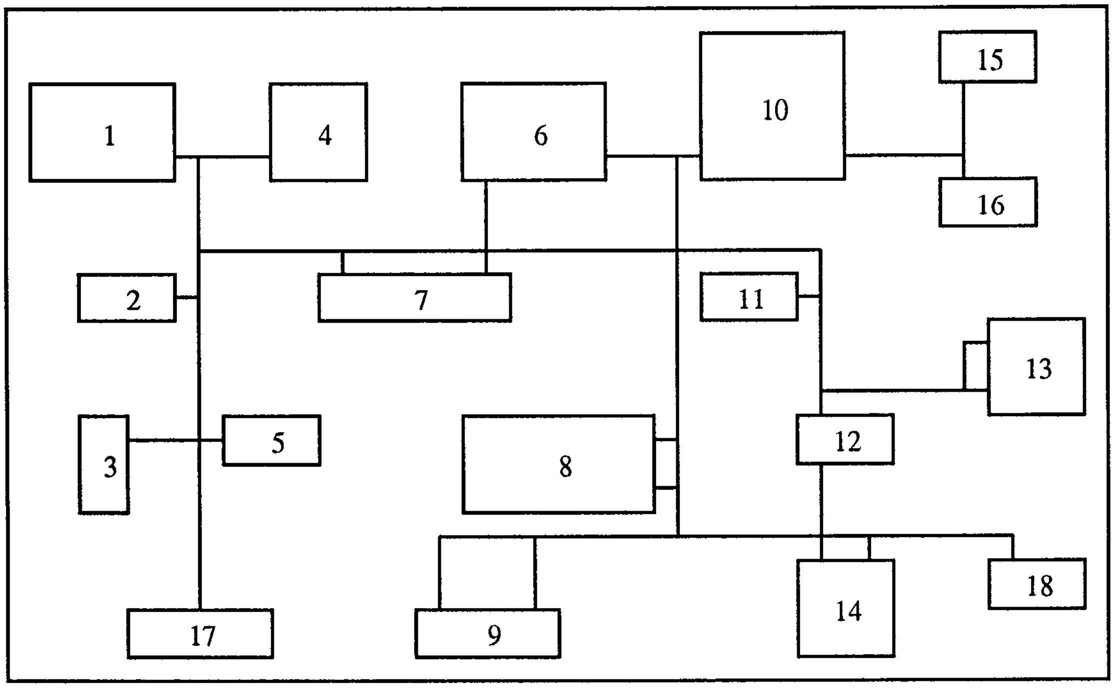

Figure 4: Inter-resource material flow paths (W_2)

charges and smaller capacities. This corresponds to a manufacturing shop where expensive automated material systems (such as Automated Guided Vehicles or linear bi-directional conveyors) are to be employed. In this case, a fixed charge of $F_{ij} = 500$ units has been assigned to each candidate arc (as in case W_2), while the capacity was set to $B_{ij} = 700$ units of flow. Again, for every commodity the variable cost was set equal to the scaled arc length (1 unit). The best configuration was obtained by FCAH and is shown in Figure 5. This solution corresponds to a total cost $Z = 87716$; i.e. fixed cost $Z_f = 49500$ and variable cost $Z_v = 38216$.

It is worth noting that some path segments are constructed to provide sufficient capacity. One example is path c-d in Figure 5, which is activated to accommodate flow which could be routed through path a-b if the capacity of this aisle had been greater. In addition, several flows are not routed through shortest paths of the original graph since the high fixed cost discourages the activation of all the required arcs, or because of capacity constraints.

Table 6 summarizes the cost metrics for the three configurations. It shows the resulting values of the cost metrics under different fixed charges and capacities. The first row provides the values of the fixed and variable costs for configuration W_1, the second row provides the

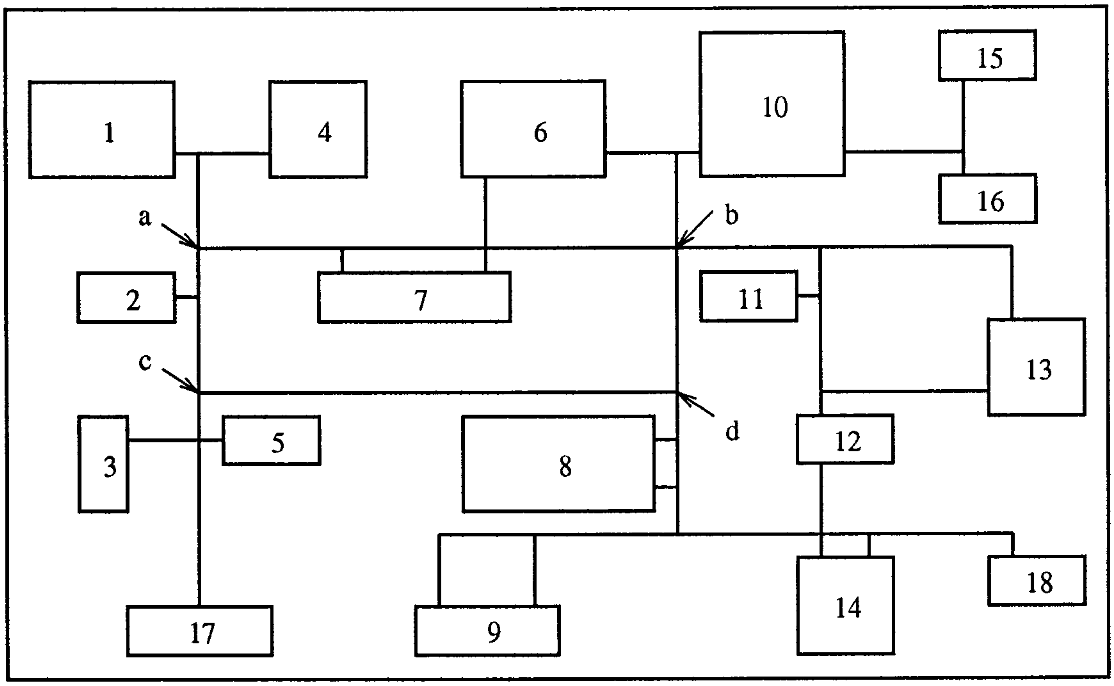

Figure 5: Inter-resource material flow paths (W_3)

corresponding metrics for configuration W_2, and the third row the ones for configuration W_3.

Table 6: Fixed and variable cost for configurations W_1, W_2 and W_3

	Z_f ($F_{ij} = 100$)	Z_f ($F_{ij} = 500$)	Z_v	B_{ij}
W_1	10800	54000	36071	1500
W_2	8500	42500	39841	3779
W_3	9900	49500	38216	700

5 Conclusions

The design of the material handling flow paths is an important part of the overall facility design problem. A multi-commodity fixed charge capacitated network design model has been developed to determine a material flow network that is inexpensive to construct and operate. This model offers significant advantages over existing ones, since it incorporates critical practical concerns. However, the problem is computationally complex and, thus, optimal solutions cannot be obtained for medium-to-large problems. Two effective heuristic solution techniques are proposed, and extensive computational tests have been performed. The latter show that the quality of the heuristic solutions is satisfactory (on average within 8% of the lower bound obtained by a dual ascent procedure). The solution approach has been employed to re-design the material flow paths of an existing manufacturing facility.

The proposed flow path design method can be employed to concurrently design the shop layout and the material handling aisles. Efficient combinatorial optimization techniques, such as simulated annealing or genetic algorithms, can be used to determine a near-optimal placement of resources. For each candidate shop configuration, the proposed method can be used to design the material flow paths and thus obtain a realistic value of the objective function of the overall problem. At the end of the procedure, the solution provides both the resource locations and the material flow network.

Finally, both the layout and flow path design problems are affected by the type of material handling systems and the routing and dispatching strategies implemented. These strategies, which are evaluated by solving the *control problem*, include unloaded moves between resources and thus alter the from-to material flow matrix. In order to exploit the relationship between these three problems, two approaches are suggested in order to evaluate layouts:

- iterate between the flow path design problem and the control problem to obtain a good combination of solutions
- generate a set of strategies for the control problem, determine a flow path design for each strategy and the given layout, and select the best combination

However, since three computationally complex problems are to be addressed simultaneously, significant work needs to be done to integrate the solution approaches.

Acknowledgments

This work was supported by the Institute for Systems Research of the University of Maryland under Grant # NSFD CD 8803012.

Appendix A

In this appendix, the Iterative Dual Ascent algorithm of Herrmann *et al.* [8] is presented. It is an extension of the labeling method of Balakrishnan *et al.* [1] to the capacitated fixed-charge network design problem. It is motivated by the observation that increases in the variable cost of bottleneck arcs (the capacity of which is exceeded in the initial primal solution) will divert flow to additional paths in the network. The algorithm iteratively updates the variable cost of bottleneck arcs, thus increasing the lower bound, generates a sequence of lower bounds and terminates with a primal feasible solution.

The basic idea of the Iterative Dual Ascent algorithm stems from the shortest path property of the dual solution derived by the labeling method. Since the active arcs (in $\bar{A}_o$) which form the shortest paths, cannot accommodate all of the commodities, flow must be directed to new arcs. This can be achieved by increasing the cost of the current shortest paths in order to create new origin-destination paths that satisfy the shortest path property.

The underlying mechanism for the required flow diversion is straightforward. Bottleneck arcs, $\{i,j\} \in \bar{A}_o$, are identified and their variable costs c_{ij}^k are increased for some commodity $k \in K$ in an attempt to alter the origin-destination shortest paths. This variable cost increase can be thought of as a penalty for violated capacity constraints. Subsequently, by re-applying the labeling method, the slacks of additional arcs are reduced to zero. Consequently, by iteratively implementing the labeling method and increasing the cost of bottleneck arcs, we eventually obtain a set of active arcs such that the capacitated problem has a feasible solution.

Algorithm IDA

Step 1 : initialization of dual variables and slacks

$q \leftarrow 1$

$w_{ij}^k \leftarrow 0 \ \forall (i,j) \in A, k \in K$

$s_{ij} \leftarrow F_{ij} \ \forall \{i,j\} \in \bar{A}$

$u_i^k \leftarrow$ shortest path from $O(k)$ to node i, $\forall i \in N, k \in K$

$Z_D^o \leftarrow \sum_{k \in K} u_{D(k)}^k$

Step 2 : initialization of labeled/unlabeled arc sets

$N_2(k) \leftarrow \{D(k)\}$, $\forall k \in K$

$N_1(k) \leftarrow N \setminus \{D(k)\}$, $\forall k \in K$

Set $CND = \{k \in K : O(k) \in N_1(k)\}$

Step 3 : evaluation of δ-increase

Select $k \in CND$

Set $A(k) = \{(i,j) \in A : i \in N_1(k), j \in N_2(k)\}$

Set $A'(k) = \{(i,j) \in A(k) : c_{ij}^k + w_{ij}^k - (u_j^k - u_i^k) = 0\}$

Calculate $\delta_1 = min\{s_{ij} : (i,j) \in A'(k)\}$

Calculate $\delta_2 = min\{c_{ij}^k + w_{ij}^k - (u_j^k - u_i^k) : (i,j) \in A(k) \setminus A'(k)\}$

Set $\delta \leftarrow min\{\delta_1, \delta_2\}$

Step 4 : dual variable update and node labeling

$w_{ij}^k \leftarrow (w_{ij}^k + \delta)$, $s_{ij} \leftarrow (s_{ij} - \delta)$, $\forall (i,j) \in A'(k)$

$u_t^k \leftarrow (u_t^k + \delta)$, $\forall t \in N_2(k)$ and $Z_D^o \leftarrow (Z_D^o + \delta)$

Update sets $N_1(k)$ and $N_2(k)$ by labeling (at most) one node:

If $\delta = \delta_1$, $s_{ij} = 0$ for some $(i,j) \in A'(k)$ set $N_1(k) \leftarrow N_1(k) \setminus \{i\}$ and $N_2(k) \leftarrow N_2(k) \cup \{i\}$

Set $CND \leftarrow CND \setminus \{k\}$; if $CND \neq \emptyset$ go to *Step 3*

If $O(k) \in N_2(k)$, $\forall k \in K$, set $Z_D^q = Z_D^o$ and go to *Step 5*

Otherwise set $CND = \{k \in K : O(k) \in N_1(k)\}$ and go to *Step 3*

Step 5 : feasibility check on zero slack arcs

Set $\bar{A}_o^q = \{\{i,j\} \in \bar{A} : s_{ij} = 0\}$

Set $G'_q = (N, A_o^q)$

if $Q_{G'_q} \neq \emptyset$ and $q \neq 1$ set $Z_{lb}^q = Z_{lb}^{q-1}$ and go to *Step 7*

if $Q_{G'_q} \neq \emptyset$ and $q = 1$ set $Z_{lb}^q = Z_D^o$ and go to *Step 7*

if $Q_{G'_q} = \emptyset$ identify $\tilde{A} = \{\{i,j\} \in \bar{A}_o^q$ that violate constraint (3)$\}$

Step 6 : variable cost update

Set $\phi_q = min\{s_{ij} : \{i,j\} \in \bar{A} \setminus \bar{A}_o^q\}$

Select $k_q \in K$ and $\{i_q, j_q\} \in \tilde{A}$

Set $c_{i_q j_q}^{k_q} \leftarrow (c_{i_q j_q}^{k_q} + \phi_q)$

Set $Z_{lb}^q = (Z_D^o + \phi_q)$

$q = q + 1$

Go to *Step 2*

Step 7 : termination

Output lower bound Z_{lb}^{q}

Solve the linear relaxation of $\mathbf{CFP}(G'_q)$ to obtain $x = (x_{ij}^{k})$

Set $y_{ij} = 1, \forall \{i,j\} \in \bar{A}_o^q : x_{ij}^k > 0$

Output primal feasible solution (x, y)

References

[1] Balakrishnan, A., Magnanti, T.L. and Wong, R.T., 1989. A dual-ascent procedure for large-scale uncapacitated network design, *Operations Research*, vol. 37, no. 5, pp. 716-740.

[2] Chhajed, D., Montreuil, B. and Lowe, T.J., 1992. Flow network design for manufacturing systems layout, *European Journal of Operational Research*, vol. 57, no. 2, pp. 145-161.

[3] Egbelu, P.J. and Tanchoco, J.M.A., 1986. Potentials for bi-directional guide path for automated guided vehicle based systems, *International Journal of Production Research*, vol. 24, no. 5, pp. 1075-1097.

[4] Gaskins, R.J. and Tanchoco, J.M.A., 1987. Flow path design for automated guided vehicle systems, *International Journal of Production Research*, vol. 25, no. 5, pp. 667-676.

[5] Goetz, W.J.Jr. and Egbelu, P.J., 1990. Guide path design and location of load pick-up/drop-off points for an automated guided vehicle system, *International Journal of Production Research*, vol. 28, no. 5, pp. 927-941.

[6] Heragu, S.S., 1992. Recent models and techniques for solving the layout problem, *European Journal of Operational Research*, vol. 57, no. 2, pp. 136-144.

[7] Heragu, S.S. and Kusiak, A., 1991. Efficient models for the facility layout problem, *European Journal of Operational Research*, vol. 53, no. 1, pp. 1-13.

[8] Herrmann, J.W., Ioannou, G., Minis, I. and Proth, J.M. A dual ascent approach to the fixed-charge capacitated network design problem, submitted to the *European Journal of Operational Research*, also available as TR 94-29, Institute for Systems Research, University of Maryland at College Park.

[9] Johnson, D.S., Lenstra, J.K. and Rinnooy, H.G., 1978. The complexity of the network design problem, *Networks*, vol. 8, pp. 279-285.

[10] Kaspi, M. and Tanchoco, J.M.A., 1990. Optimal flow path design of unidirectional AGV systems, *International Journal of Production Research*, vol. 28, no. 6, pp. 1023-1030.

[11] Kim, H.K. and Tanchoco, J.M.A., 1991. Flow path design of fixed-path material handling systems, *ASME Annual Meeting: Planning and Control of Material Handling Systems*, MHD-vol. 1, pp. 33-40.

[12] Kim, H.K. and Tanchoco, J.M.A., 1993. Economical design of material flow paths, *International Journal of Production Research*, vol. 31, no. 6, pp. 1387-1407.

[13] Krishnamurthy, N.N., Batta, R. and Karwan, M.H., 1993. Developing conflict free routes for Automated Guided Vehicles, *Operations Research*, vol. 41, no. 6, pp. 1077-1090.

[14] Magnanti, T.L. and Wong, R.T., 1984. Network design and transportation planning: models and algorithms, *Transportation Science*, vol. 18, no. 1, 1-55.

[15] Maxwell, W.L. and Wilson, R.C., 1981. Dynamic network flow modeling of fixed path material handling systems, *AIIE Transactions*, vol. 13, no. 1, pp. 12-21.

[16] Minis, I., Harhalakis, G. and Jajodia, S., 1990. Manufacturing cell formation with multiple, functionally identical machines, *Manufacturing Review*, vol. 3, no. 4, pp. 252-261.

[17] Minoux, M., 1989. Network synthesis and optimum network design problems: models, solution methods and applications, *Networks*, vol. 19, pp. 313-360.

[18] Montreuil, B. and Ratliff, H.D., 1989. Utilizing cut trees as design skeletons for facility layout, *IIE Transactions*, vol. 21, no. 2, pp. 136-143.

[19] Proth, J.M. and Souilah, A., 1992. Near-optimal layout algorithm based on simulated annealing, *International Journal of Systems Automation: Research and Applications*, vol. 2, pp. 227-243.

[20] Sharp, G.P. and Liu, F.F., 1990. An analytical method for configuring fixed-path, closed-loop material handling systems, *International Journal of Production Research*, vol. 28, no. 4, pp. 757-783.

[21] Sinriech, D. and Tanchoco, J.M.A., 1991. Intersection graph method for AGV flow path design, *International Journal of Production Research*, vol. 24, no. 5, pp. 1725-1732.

[22] Venkataramanan, M.A. and Wilson, K.A., 1991. A branch-and-bound algorithm for flow-path design of automated guided vehicle systems, *Naval Research Logistics*, vol. 38, pp. 431-445.

[23] Wilhelm, M.R. and Evans, G.W., 1987. State-of-the-art modeling techniques for AGV systems design and operation, *Material Handling Focus '87*, Material Handling Research Center, Georgia Institute of Technology, Atlanta, GA 30332.