An Integrated Design Algorithm for Detailed Layouts Based on the Contour Distance

Jae-Gon Kim and Marc Goetschalckx
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205

Abstract

This paper presents an integrated design approach for the facility layout problem. It develops a method for the concurrent determination of the block layout, the locations of departmental input and output (I/O) points using the contour distances between the I/O points, and the material flow paths between the I/O points. Results of computational experiments show that the performance of this integrated algorithm compares favorably with those of algorithms using a sequential approach.

Keywords

Layout design, block layout, detailed layout, flow path design, I/O point location, contour distance.

1. Introduction

The facility layout problem is concerned with the conceptual design of the physical enclosure of manufacturing or service systems. The material handling costs of these systems are directly affected by the arrangement or layout of the facility. A commonly used objective in the facility layout design is to minimize the total transportation distance (TTD), which is defined as the sum of material flows between two departments weighted by material transportation distances along the flow paths (aisles) from output (pick-up) points of a department to input (drop-off) points of another department. The three principal and interdependent design decisions in the facility layout design problem are: 1) the determination of the shapes and locations of departments within the facility, which is called the conceptual block layout problem; 2) the determination of the locations of the input and output (I/O) points on the boundary of each department; and 3) the design of the flow paths or aisles that connect these I/O points. Although these three problems are clearly closely related, traditionally they have been solved separately in a sequential manner because of the computational intractability of the integrated design problem.

Usually, the locations of the I/O points and the flow paths are determined for a given block layout after the block layout has been generated based on rectilinear centroid-to-centroid distances. However, such a sequential single-pass approach may generate significantly suboptimal designs. Therefore, the development of an integrated approach to determine a block layout, locations of I/O points and flow paths concurrently is required to generate better designs and to judge the performance of sequential heuristics and optimal algorithms.

1.1 Abbreviated Review of Current Methodology

There exist a large number of optimal and heuristic algorithms for the generation of a conceptual block layout (the block layout problem). Solving the block layout problem to optimality is computationally very demanding since the problem has been shown to belong to the class of NP-Hard problems. Typically, the problem is formulated as a mixed integer non-linear programming model, which cannot be solved in a reasonable amount of time for problems of realistic size. Hence, most design methods have focused on developing heuristic algorithms. See Kusiak and Heragu (1987) and Meller and Gau (1996) for extensive surveys on design algorithms for the block layout problem.

In the block layout problem, it is commonly assumed that I/O points of each department are located at the centroid of that department and the centroid-to-centroid rectilinear distance is used to calculate the material flow distance between the departments. These assumptions are clearly violated in most real-life layouts. It is much more realistic to assume that the I/O points of departments are located on the boundaries of the departments and that materials are moved along flow paths connecting them. Benson and Foote (1997) present a genetic algorithm for determining locations of I/O points when material transportation occurs in aisles. They use the shortest aisle distances between the I/O points to calculate the TTD. Kim and Kim (1999) present a procedure to obtain finite candidate positions for each I/O point and suggest a branch and bound algorithm to find optimal locations of I/O points. In their study, it is assumed that materials are moved along given flow paths or boundaries of departments.

There exist a limited number of design algorithms for the flow path design problem. Wu and Appleton (2000) suggest a genetic algorithm to find the shortest paths for all material flows using the contour distances between I/O points for a given block layout with a slicing structure. A layout is said to have a slicing structure when the boundaries of all departments can be constructed by successive cuts of the remaining layout area, where each individual straight-line cut completely separates the remaining area. Such cuts are also called guillotine cuts.

Integrating more than two components of the three layout components (block layout problem, I/O point location problem, and flow path design problem) in an integrated design procedure has rarely been considered in the literature because of the computational difficulties. Palliyil and Goetschalckx (1994) present an MIP model for simultaneously determining the flow paths and locations of I/O points on a given block layout with the objective of minimizing a sum of the material transportation cost and fixed setup cost for design of material flow networks. For generating a block layout and flow paths simultaneously, Ho and Moodie (2000) propose a constructive heuristic algorithm based on linear programming. In their algorithm, however, it is not guaranteed that a final layout is compact, i.e. the layout may have empty spaces on the floor plan. Wu and Appleton (2002) suggest a method for representing a slicing block layout and its aisle structure using strings and determine a block layout and aisle structure simultaneously using a genetic algorithm. Norman et al. (2001) embed a heuristic algorithm for determining locations of I/O points into the genetic algorithm of Tate and Smith (1995) to obtain a slicing block layout based on the contour distance. They assume that each department can have an unlimited number of I/O points for easy calculation of the contour distances between the I/O points and then determine an appropriate number of I/O points for each department.

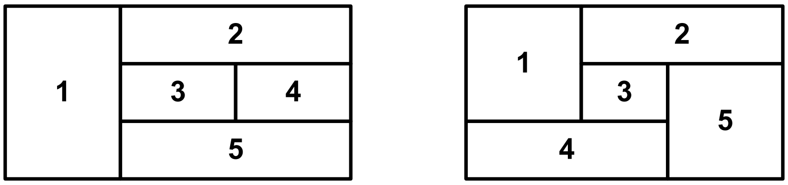
2. Development of a Formulation for Detailed Layout Design based on Contour Distances

In this paper, we present an integrated approach to obtain a block layout, the locations of I/O points, and the flow paths simultaneously. The sequence-pair structure is used to represent the relative positions of departments in a block layout, which was originally proposed by Murata et al. (1996) for the physical design of VLSI layouts. A linear programming model is used to generate a block layout from a sequence-pair and the locations of I/O points are then determined using three heuristic methods based on the solution of the all-pairs shortest path problem. These methods are embedded into a simulated annealing algorithm to obtain a high-quality layout design considering the block layout and locations of the I/O points and using the contour distance. Note that the aisle design problem is not explicitly considered in this study. However, a feasible set of flow paths can be easily constructed based on the shortest paths along department boundaries between the I/O point locations.

2.1 Conceptual Block Layout Based on Rectilinear Distances

It is very difficult to develop a concise and complete mathematical formulation of the facility layout problem, because a condensed mathematical expression for the contour distance between I/O points does not exist when the block layout (and the contours) are not given but are decision variables. Therefore, an initial location of the I/O points is based on an initial formulation that uses the rectilinear distances instead of the contour distances and that assumes that the I/O points of a department can be placed at the interior of the department. The I/O point locations are then later improved using heuristic methods to obtain feasible locations of the I/O points on the perimeter and using contour distances. The initial facility layout problem can be formulated as a mixed integer non-linear program (MINLP).

A sequence-pair, $\Gamma = (\Gamma^+ ; \Gamma^-)$, is a pair of sequences (Γ^+ and Γ^-) of indexes of departments. When a block layout is given, the relative positions of departments in the block layout can be represented a sequence-pair.



$$\Gamma^+ = (1,2,3,4,5) \qquad \Gamma^- = (1,5,3,4,2) \qquad \Gamma^+ = (1,2,3,4,5) \qquad \Gamma^- = (4,1,3,5,2)$$

Figure 1. Block Layouts with Slicing and Non-slicing Structure

Murata et al. (1996) proved that for an arbitrary Γ there always exist block layouts whose topology is represented by Γ, if the floor area is infinite and the aspect ratio constraints are ignored. Note that there exist an infinite number of block layouts whose topology is represented by an arbitrary sequence-pair, since there are an infinite number of ways to place departments on a continuous plane while maintaining the relative positions among them. However, for every block layout there exists a unique sequence-pair that represents the topology of the block layout, if, without a loss of generality, when department i is above (below) department j and at the same time to the left (or right) of department j, department i is considered to be above (below) department j.

We use the polyhedral outer approximation method presented by Goetschalckx (1998) and Chen and Kuh (2000) to linearize the quadratic department area constraint. The tangential supports are generated at different points on the area curve and the area bounded by these inequality constraints forms an approximated feasible region for the area constraint of department i.

In this study, a simulated annealing (SA) algorithm is used to find a high-quality sequence-pair that gives a feasible layout (a block layout and locations of I/O points) with the smallest TTD. In the SA algorithm, a sequence-pair is generated as a solution. The linear model is then used to generate a layout from the sequence-pair and the enhanced improvement heuristics are used to improve locations of I/O points. The TTD obtained based on those I/O point locations is then used as the evaluation score of the sequence-pair.

2.1 Location of I/O Points on the Boundary of Departments

Next heuristic methods for adjusting the locations of I/O points using the contour distances between the I/O points are presented. At first, a finite set of candidate positions for each I/O point is generated. Then the size of the set is reduced by eliminating dominated candidate positions from the set. Finally, three heuristic methods are developed that improve the locations of I/O points using iterative procedures.

Even though the I/O points of a department can be placed at any location on the boundaries of the department, it is sufficient to consider only the intersections of the boundaries of the department as candidate positions for the I/O points (Kim and Kim, 1999). Here, intersections are defined as the points at which horizontal and vertical boundaries of two departments meet plus the four corner points of each department (if these corner points are not already included). Among the candidate positions for each I/O points, some can be dominated by others. Kim and Kim (1999) developed conditions when an intersection is dominated by another intersection and can thus be eliminated from the candidate positions for an I/O point.

For determining location of I/O points, three improvement heuristic methods are proposed: a sequential improvement method, an iterative improvement method, and an extended iterative improvement method. In all three methods, the initial locations of I/O points are obtained from the LP solution and are based on the rectilinear distance. If an I/O point in the LP solution is located at one of its candidate positions, then this candidate position becomes the initial position of the I/O point. Otherwise, the I/O point is moved to its nearest candidate position on the contour, where nearest is based on the rectilinear distance norm. Ties are broken arbitrarily. The initial locations of I/O points are then improved with one of the three methods.

In the sequential method, the initial locations of I/O points are improved by moving I/O points sequentially (one-at-a-time) to better candidate positions in such a way that the decrease in the TTD is maximized. In iterative method, the positions of all input points are changed simultaneously and then the positions of all output points are changed simultaneously in an iterative manner to improve locations of I/O points. When positions of input points are given, the best position of an output point of a department can be determined independently of the location of the output points of other departments, since there is no material flow between the output points of different departments. That is, the best position for the input point for each department can be obtained by evaluating all candidate positions using the given information of current positions of the output points.

In the previous two methods, no further improvement of the solution (the locations of I/O points) occurs once the solution reaches a local optimum. The following method is designed to escape from a local optimum and to search for an alternative local optimum. The method escapes from a local optimum by moving one input (or output) point to its worse candidate position and then finding a new local optimum by moving output (or input) points and input (or output) points to their best positions by using the iterative improvement method.

3. Computational Experiments

3.1 Heuristic Location of I/O Points on the Boundary of Departments

In this study, the three heuristic methods of determining locations of I/O points are compared first with the optimal location algorithm of Kim and Kim (1999). Then, the developed SA algorithm is compared with algorithms based on the sequential approach in which a block layout is generated first without consideration of the locations of I/O points and then the locations of I/O points are determined for the given block layout. The tests were executed on a personal computer with a 700MHz Pentium-III processor and CPLEX 6.0 was used to solve the linear programs in the algorithms.

Table 1. Performance of the Heuristic I/O Point Location Methods

Number of	Average optimality gap (%)			
departments	Initial†	SIM	IIM	EIIM
10	4.59	0.491	0.477	0.045
15	6.40	0.39	0.377	0.030
20	3.50	0.107	0.142	0.014
30	2.62	0.08	0.08	0.009
40	2.28	0.046	0.046	0.011

†Initial locations obtained by modifying the LP solutions in the heuristic methods

Table 2. Average CPU Times of the Heuristic I/O Point Location Methods

Number of	Average CPU time (seconds)			
departments	Optimal†	SIM	IIM	EIIM
10	0.052	4.66×10^{-5}	7.31×10^{-5}	7.81×10^{-4}
15	0.120	1.68×10^{-4}	1.61×10^{-4}	1.76×10^{-3}
20	0.282	2.55×10^{-4}	2.10×10^{-4}	1.11×10^{-3}
30	1.013	8.76×10^{-4}	4.40×10^{-4}	2.03×10^{-3}
40	2.169	1.98×10^{-3}	7.56×10^{-4}	2.96×10^{-3}

†Optimal locations obtained by using the branch and bound algorithm of Kim and Kim (1999)

In Table 1, the average optimality gaps of the three heuristic methods are very close to 0, which means that these methods, especially EIIM, found optimal or good sub-optimal locations in most of cases, and the computation times of the methods are much shorter than that of the optimal algorithm of Kim and Kim (1999) in Table 2.

3.1 Integrated versus Sequential Detailed Layout Design

The SA algorithm describe above is compared with algorithms based on a sequential approach. For the algorithms based on the sequential approach, a block layout is generated using one of the existing block layout generation algorithms and then the optimal locations of I/O points are determined using the optimal location algorithm of Kim and Kim (1999). In this study, the SA algorithm of Kim and Kim (1998) and the GA algorithm of Tate and Smith (1995) are used for generating block layouts.

For the comparison between the integrated and the sequential approach, the five test problems generated in section 6.1, the 10-department problem of Van Camp et al. (1991) and the 20-department problem of Armour and Buffa (1963) were used. In all the test problems except for the test problem of Van Camp et al. (1991), the range of aspect ratio was set to (0.25, 4.0). For the test problem of Van Camp et al. (1991), we use the original data for the range of the aspect ratio of Van Camp et al. (1991).

For each problem, 10 replications were made and the results are shown in Tables 3 and 4. Table 3 shows the minimum and average TTD values and the standard deviation of the TTD values for each problem. For all the problems, INTE-KG showed the best minimum and average TTD values and its performance was much better than SEQ-KK and SEQ-T especially for the small-sized problems in terms of the three comparison measures. Table 4 shows the relative deviation index (RDI) of each algorithm for each problem and the average computation times of the three algorithms for each problem.

Table 3. Comparison of the integrated approach and the sequential approach

Problem	Algorithm								
	INTE-KG			SEQ-KK			SEQ-TS		
	min	avg	std	min	avg	std	min	avg	std
10_Van Camp	6461.4	8439.2	1423.1	11627.1	14726.6	2267.6	9530.5	17222.7	2702.8
20_Armour and Buffa	2699.9	3018.7	121.5	3044.4	3501.3	324.1	3457.5	3593.2	104.5
10_random	976.19	1042.1	62.4	1241.3	1332.4	168.7	infeasible†		
15_random	1961.7	2233.6	187.3	2737.8	2872.3	255.9	2945.2	3065.5	39.2
20_random	8209.1	9192.1	685.1	9093.5	10302.0	622.8	9535.3	10256.7	851.7
30_radom	28376.9	29606.3	1005.1	29295.9	30403.1	736.9	31803.9	33221.7	972.7
40_random	68430.0	70119.8	1140.5	69188.8	70641.8	1025.6	79006.2	83045.5	2296.4

†SEQ_TS did not find a feasible layout for 10_random.

Table 4. Relative deviation indexes and computation times

Problem	Algorithm						Average CPU times (seconds)
	INTE_KG		SEQ-KK		SEQ-TS		
	RDI-M	RDI-A	RDI-M	RDI-A	RDI-M	RDI-A	
10_Van Camp	0.00	0.00	0.80	0.75	0.47	1.04	40.0
20_Armour and Buffa	0.00	0.00	0.13	0.16	0.28	0.19	726.5
10_random	0.00	0.00	0.27	0.28	–		65.1
15_random	0.00	0.00	0.40	0.29	0.50	0.37	270.8
20_random	0.00	0.00	0.11	0.12	0.16	0.12	852.4
30_random	0.00	0.00	0.03	0.03	0.12	0.12	3766.1
40_random	0.00	0.00	0.01	0.01	0.15	0.18	15622.1

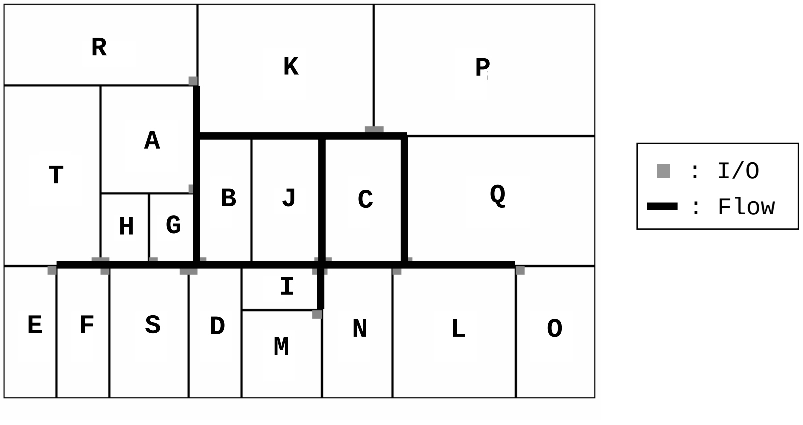


Figure 5. The Best Layout Obtained by INTE-KG for 20_Armour and Buffa

5. Conclusions and Future Research

The computational experiments showed that the suggested block layout generation method and the I/O point location methods are computationally very efficient. This new algorithm is also very effective when compared to algorithms that are using a sequential approach.

In the algorithm presented here, the flow paths are determined by the shortest path along the perimeter of departments between the I/O points, without consideration of the area and cost implications for the construction of the flow paths. However, more realistic layouts may be generated if the flow paths are determined considering the capacities of flow paths, the smoothness of material flows, and the costs and area requirements for the constructing of flow paths.

Acknowledgements

This work has been funded in part by the W. M. Keck Foundation and the Ford Motor Company.

References

[1] Armour, G. C. and Buffa, E. S., 1963, A heuristic algorithm and simulation approach to relative location of facilities. Management Science, 9, 294-309.

[2] Benson, B. and Foote, B. L., 1997, DoorFAST: A constructive procedure to optimally layout a facility including aisles and door locations based on an aisle flow distance metric. International Journal of Production Research, 35, 1825-1842.

[3] Chen, P. and Kuh, E. S., 2000, "Floorplan sizing by linear programming approximation," Proc. 37th Design Automation Conference, 468-471.

[4] Chittratanawat, S. and Noble, J. S., 1999, An integrated approach for facility layout, P/D locations and material handling system design. International Journal of Production Research, 37, 683-706.

[5] Goetschalckx, M., 1998, Models and algorithms for block layout design with departmental shape constraints. Proceedings of the 1998 Progress in Material Handling Research Conference, Phoenix, AZ, Material Handling Institute, Charlotte, NC.

[6] Ho, Y-C. and Moodie, C. L., 2000, A hybrid approach for concurrent layout design of cells and their flow paths in a tree configuration. International Journal of Production Research, 38, 895-928

[7] Kim, J-G. and Kim, Y-D., 1998, A space partitioning method for facility layout problems with shape constraints. IIE Transactions, 30, 947-957.

[8] Kim, J-G. and Kim, Y-D., 1999, A branch and bound algorithm for locating I/O points of departments on the block layout. Journal of the Operational Research Society, 50, 517-525.

[9] Kim, J-G. and Kim, Y-D., 2000, Layout planning for facilities with fixed shapes and I/O points. International Journal of Production Research, 38, 4635-4653.

[10] Kusiak, A. and Heragu, S. S., 1987, The facility layout problem. European Journal of Operational Research, 29, 229-251.

[11] Meller, R. D. and Gau, K-Y., 1996, The facility layout problem: recent and emerging trends and perspectives. Journal of Manufacturing Systems, 15, 351-366.

[12] Montreuil, B., 1987, Integrating design of cell layout, input/output I/O point configuration, and flow network of manufacturing systems. In: Liu, C.R., Requicha, A. and Chandrasekar, S (eds). Intelligent and Integrated Manufacturing Analysis and Synthesis. Ped-Vol. 25, ASME: New York, pp 315-326.

[13] Murata, H., Fujiyoshi, K. and Kajitani, Y., 1996, VLSI module placement based on rectangle-packing by the sequence-pair. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 15, 1518-1524.

[14] Norman, B. A., Arapoglu, R. A. and Smith, A. E., 2001, Integrated facilities design using a contour distance metric. IIE Transactions, 33, 337-344.

[15] Palliyil, G. and Goetschalckx, M., 1994, A comprehensive model for the concurrent determination of aisles and load stations for aisle-based material handling systems. Technical Report. School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia.

[16] Tate, D. M. and Smith, E. A., 1995, Unequal-area facility layout by genetic search. IIE Transactions, 27, 465-472.

[17] Van Camp, D. J., Carter, M. W. and Vannelli, A., 1991, A nonlinear optimization approach for solving facility layout problems. European Journal of Operational Research, 57, 174-189.

[18] Wu, Y. and Appleton, E., 2002, The optimization of block layout and aisle structure by a genetic algorithm. Computer & Industrial Engineering, 41, 371-387.